A large, stylized blue letter 'A' with a white outline, positioned on the left side of the page.

July 1988

Vol. 1 N° 10

Price £1.50

Archive

The Subscription Magazine for Archimedes Users

A solid blue horizontal bar spanning the width of the page.A large, stylized blue letter 'E' with a white outline, positioned on the left side of the page.

First Word Plus Review & Programs

Using ARM BASIC Editor

Assembly Language Programming

Pipedream, Euclid, Newsmaster & Games Reviews

A Spell-Checker for all Word-Processors?

SigmaSheet Sorting Program

Printkey ScreenDumps and ScreenSaves

Sound Synthesis (3) plus More on EMR SoundSynth

It just keeps on coming!

Once again, thanks for all the articles and information. There seems to be a lot of interest in First Word Plus – hence all the various contributions on that.

I said last month that “it’s amazing how just enough arrives each month to fill the pages” – you must be joking! This month I’ve put virtually all the articles down to 13 point instead of 14, started articles part way down a page to save space (though not very elegant, sorry) and still I’ve had to leave some stuff over until next month – a major article and program about a floating point assembler and an article about using special characters & different alphabets. Also, I’ve had to put several programs that were too long to list in the magazine, onto the monthly program disc. I don’t like doing that because it looks as if I’m trying to force people to buy the program disc, but it’s your fault – you’re sending in too much good material – but please don’t stop!

If you keep sending in the material, I’ll keep producing a jam-packed magazine and, presumably, that will keep everyone happy!

Finally, I have found a verse that sums things up for me just at present: Have a look at Ecclesiastes chapter 12, verse 12 which says,

“Of making many books there is no end, and much study wearies the body”!

Have a good summer holiday!

A handwritten signature in cursive script that reads "Paul B." The signature is written in dark ink and is positioned centrally below the main body of text.

Archive

Volume 1 • Nº 10 • July 1988

Contents

Hardware & Software Available ...2	More Mode Conversions23
Comment Column.....3	View to First Word Plus24
Readers' Comments3	Using the ARM BASIC Editor ...30
Matters Arising4	A Spell-Checker for all WP's?....34
Help!!!.....5	Pipedream Review35
Information about Monitors5	Contact Box37
Bug or Feature?.....5	Assembly Lang. Programming ...38
Hints & Tips.....6	NewsMaster Review39
Hardware Column.....11	Sound Synthesis – Part 341
Small Ad's.....12	EMR SoundSynth44
MS-DOS Column13	Printkey Screen Dumps45
BASIC V Utilities Forum16	More Software Available46
First Word Plus Review18	Games Galore!47
First Word Plus Colour Palette ...19	Euclid 3D Designer Review48
First Word Plus Auto-Dating20	SigmaSheet Sorting Program.....50
First Word Plus Notes21	Fact-File53
Wordwise to First Word Plus22	

Hardware & Software Available

• 4-slot backplanes for the 300 series!!!!

Available (by the time you get this) from Computerware, 55 Romulus Court, Brentford Dock, Justin Close, Brentford, Middlesex for only £59.95 (including fan). (£2.50 off if you quote your Archive subscription number!)

• **Buffer Podule** from SGB Computer Services (£49.50 inc.), available now, allows a single podule to be placed external to the computer. Expansion Box (Price TBA) will house up to 3 double or 5 single width podules.

• **New improved 24-pin screen dump.** Now this is what I like to hear... Abacus Training have responded to (constructive) criticism from early users of their 24-pin screen dump and have implemented some of their suggestions. The new version, will now dump all or part of the screen, has an optional left margin and allows 1, 2 or 3 passes to increase the blackness of the image, if required. To get an up-grade, early users can send £1.50 for a new disc OR send back the old disc in a Jiffy bag and get the upgrade free of charge. (For new purchasers, the price is still £5.00.)

• **Colour Dump for Canon Colour Ink Jet printer** is available now from Musbury Consultants £30 (£25 through Archive).

• **Professional CAD Software** – Oak Professional Software have produced a “fully interactive parametric CAD package” for the Archimedes. The Oak PDT (Parametric Design Tool) runs on Archimedes with 1 Mbyte or more. It has a huge range of features and is also a “low cost” package at only £295 (+VAT?). For more information, ring 0532-502615

• **Sound & Graphics Demo** from Clares – only £7.50 for a two-disc set with some VERY impressive demonstrations that show off the Archimedes to very good effect. (At that price, there's no dealer discount, so you'll have to get it directly from Clares.)


• **Facsimile for the Archimedes** from Computer Concepts. Based around Computer Concepts own 9600 bps modem podule, you can transfer data to and from group 2 and 3 FAX machines anywhere in the world. Available in three versions: FAX PACK 1, modem and software only £499+VAT, FAX PACK 2, including a 200 dpi scanner, £799+VAT and FAX PACK 3, with a flat-bed 300 dpi grey scale scanner “over £1100+VAT”. Release “summer 1988 subject to BABT approval”.

• **Archive Shareware Disc** The long-awaited(?) Archive Shareware Disc is here! The disc contains 771k of demos and other programs. These include: the new LIFE program – this has a higher resolution and runs at just about the same speed; a new version of the mandelbrot program – this includes a gallery function; a ‘European Geography’ program – all you wanted to know about the European countries with sprites to match. There are also several compacted screen shots, a beautiful mandelbrot screen, and some smaller demos. Many thanks to all the various contributors. Price £3.

• **Database of Educational Software and Addresses** from Nick Evans, mentioned in the May issue has been delayed – Nick has spent the last two months in hospital with back problems, so please be patient. Get well soon, Nick!

Review Software & Hardware Received...

Apart from reviews already written, we have received review copies of the following: Printer Buffer Module from Clares, Archimedes Kermit from Acornsoft, Floating Point Assembler from Abacus Training, Presenter (Presentation Graphics) from Linguinity, Buffer Podule from SGB Computer Services.

(See also, More Software Available, on page 46 for some software that arrived at the last minute before printing.) 

Comment Column

EUREKA!

I reckon it should be re-named Disaster! After a couple of weeks during which the Eureka! Bulletin Board worked tolerably well, it got worse and worse. The hard disc kept going wrong and causing us to lose data. In the end we could not even re-format it! We tried a second Master computer in case that was at fault. We have had various up-dates of the software (ACMB admit that there are some problems with running on the Pace Linnet that we chose to use.) We're going to give ACMB one more chance to redeem themselves but if we still have no joy, we'll have to start again from scratch with some other bulletin board software. (We tried to modify the ACMB software ourselves but gave that up as a bad job when we saw the high frequency of one dreaded keyword!)

A note from the Sysop, Carl Wright

Just a note to tell you that we are on the look out for various articles, downloads etc. for the Eureka! Bulletin Board System. If you are interested in submitting any information or downloads then either call Norwich Computer Services or if you have access to a suitable 1200/75 (V23) or 300/300 (V21) modem and some viewdata or scrolling text terminal software then you can call the Eureka! BBs any time.

If you aren't busy then you can set up your own section on the Bulletin Board, we will give you the necessary help in how to edit. You can either edit the pages of Mode 7 text and graphics whilst 'on-line' or you can edit the pages remotely and 'up-load' them to us. We have over 15 Mbytes of free space just waiting to be filled by your Archimedes programs or routines.

You will be able to get through to the board if you have a 1200/75 or 300/300 baud modem and either scrolling text or viewdata terminal software any time on (0603) 250689. If you aren't already able to access the Archive Closed

User Group then fill in the necessary questionnaire stating your Archive Subscribers number (which is on the original letter sent out when you first subscribed!) – give us a ring if you can't remember it.

Prices going up (gone up)

By the time you get this, the prices of various bits of Archimedes kit will have increased. From 1st July, the new prices will be:

	Base	Mono	Colour
A310	835	895	1055
A440	2529	2589	2749

The A305's have not changed in price, but before you rush out and buy one, the 0.5 Mbyte upgrade has gone up from £89 to £149. Add-ons have gone up too: the I/O podule has gone up from £79 to £85 and the Midi add-on from £29 to £39. However, the 2nd floppy drive stays at £125 and the 20 Mbyte hard drive at £499. (All prices are quoted ex-VAT).

(We still have two A310 colour computers at the old price which we bought for our special promotion in June. Give us a ring to see if they are still available – £1121 inc VAT (+ carriage) to the first two callers – save £90 AND get a year's free on-site servicing.) **A**

Readers' Comments

The comments expressed here do not necessarily reflect the views of the Archive magazine editorial team.

• **Operating Systems:** Lawson Wakefield writes: Am I alone in finding Arthur a touch inadequate? After some 6 years of using Beebs, I understand why it is the way it is but does it provide the right environment for developers who need to manage code running into thousands of lines? Programs to run in one Mbyte require very different facilities to those which must fit into just 32k! What do others feel?

ROMs – Mike Harrison writes:

1) Yes, ROMs are copied into the RMA before running, but RMA resident software can load data and overlays from ROMs

2) The point—remember that 128 Kbyte Eproms and ROMs are now affordable – loading 128k from disk takes time, and 128k isn't an insignificant amount to lose by having it all in the RMA. It should, for example, be fairly simple to stuff the PC and 6502 emulators into a ROM – this would mean faster booting, and fewer disks required (useful in schools)

3) The reason CC will only issue software on PAL protected ROMs is that it takes a great deal of time (i.e. money) to develop software, and it is difficult to recoup this if people illegally copy it – the benefit to users is that the price can therefore be made lower, because fewer sales will be lost through copying. I agree that it is a shame that it costs so much for the backplane plus ROM board, but I know for a fact that CC were not happy about this. The reason they developed their ROM board was so that you could have a sensible number of ROMs on each board and that you could have more than one board in each machine. For comments on the relative merits of ROM boards, see issue 7, p 44.

SigmaSheet versus InterSheet

Those of us who cannot afford hard discs have reason to be grateful for the ability to put software in ROM. Having got used to the idea of just typing, for example, *Isheet on the BBC Micro to get that application going, it is a continual source of irritation to me to have to look around for my SigmaSheet disc, put it in and wait for it to load. I have now decided to use Inter-Sheet in ROM, in preference to Sigma-Sheet. The re-calculation time of both is so fast as to be irrelevant and the convenience of being able to specify box locations by moving the cursor and pressing <copy> makes Sigma-Sheet's 'blind' entry of box numbers seem prehistoric. If you have a bad memory for letters and figures, as I do, you have to write them down on a piece of paper before initiating the copy

function. On InterSheet, I can do much of the work without even thinking in terms of box numbers; I just think in terms of "copy the data from this box to this box", moving the cursor over the sheet, "and put it in the area from this box to this box" – much less prone to error than "copy B23,G45:N46,R68". **A**

Matters Arising

• **Logistix** – The reviewer, Mark Sealey, had problems changing drives, but it is easy (when you know how!) – use the Utilities command. Also, contrary to what Mark said, there is a 24-pin printer driver – for the LQ1500 – which also works on (amongst others?) the LQ500.

One negative for Logistix is that it is rather unforgiving if you try to enter formulae of 90 characters or more. If you exceed the 89 character limit, either on entering or editing, it doesn't warn you, it just gives you an empty cell! So make sure you have a copy of your longer formulae in case you lose them.

One 'feature' of Logistix is that numbers between 1 and -1 are not printed with a leading zero. The trouble is that .3 can be very easily mistaken for 3, especially if you are photocopying the resultant output, whereas 0.3 is much less ambiguous.

One bug (definitely NOT a feature) is that if, by mistake, you enter: +IF("abc"=0,1,0) into any cell, it crashes completely!

Acorn say they know about the bug and "as yet we have no solution to it" but that it will be dealt with "when Logistix is next updated".

• **PC Emulator up-grade.** Acorn have pointed out that the information we gave last month was incorrect. If you have already paid £15 for your up-grade to 1.09, you do not have to pay another £15 for the up-grade to 1.20. Philip Colmer assured us that as long as people mention the fact they they got the 1.09 as an up-grade, the 1.20 will be free. Those of you who have paid two lots of £15, I suggest you send your receipts to Philip and see if you can get a £15 refund! **A**

Help!!!

• **R E Boldero** of Kings Lynn asks if there is any way to program a 310M so that at power-up it will print up a message, depending on the time of day: "Good evening. How can I assist you?" or "Good morning", or "Good afternoon".

Help offered

• **Hard disks do's and don't's:** From comments we've had so far, they seem to be reliable, but it is worth getting into the habit of parking the heads by using *BYE before you switch off, and never try to move the machine while it is switched on. The Welcome Guide says there is a utility for backing up the hard drive but doesn't tell you how to do it.

A useful suggestion from one reader is to have a directory called :4.\$tmp which is always cleared by a boot file when you start up. The idea is to use it for temporary files which would otherwise lie around on the disc until you got round to a very time consuming weeding out process.

• **Econet.** Felix Andrew is currently working on a small file-server for Dulwich College to link a couple of Archimedes and one or two BBC's. He has also developed a Password Editor Program for the S.J. Research file-server (HDFS). If you are interested in either of these, get in contact with him at 19 Burbage Road, Dulwich, London SE24 9HJ. **A**

Monitor Information

• **Mike Harrison** writes... I've used the original **NEC Multisync**, and the **Taxan**. The nice thing about the Taxan is that it has separate geometry adjustments for each scanning speed, and so the horizontal shift can be corrected without affecting modes 0..17. The problem is that there are so many adjustments that it can take a bit of juggling to get it right – at least an hour! Comp-

ared to the NEC, there are 3 things I don't like:

1) It seems to run quite a bit hotter than the NEC
2) as mentioned elsewhere, the brightness isn't wonderful – especially annoying with the PC emulator, which insists on using a non-changeable palette of dark grey on black!

3) (Only a disadvantage if you are interested in video work and want to use interlaced displays) the Taxan freaks out when displaying interlaced pictures – there is a severe horizontal judder.

My only criticism of the NEC is lack of width.

NEC do a monochrome multisync (the GS) for about 160 quid – does anyone know if this will work on the Archimedes? – if it will, it will be a wonderful cheap solution for DTP and WP applications. I'm told that NEC say it won't work, but that was based on the specs – it would be useful if someone could try it and confirm one way or the other.

• Various comments have come in about the **Fujitsu multi-sync monitor** that Viglen were doing, all saying how good it is and what good value. Surely there must be some entrepreneur out there who could do something for us Archimedeans who can't afford £500 for a multisync??!! **A**

Bug or Feature?

If you have a program which chains another program on the same disc and you try to run the first program from the desktop, it will only work if the disc is in drive 0. If the disc is in drive 1, clicking on the program icon runs the first program OK, but when it tries to chain the next program it looks on drive 0. I think that it is stretching it a bit to call this a 'feature' unless anyone can think of a way of allowing a program to detect which drive it had been loaded from. This bug came to light when trying to run 1stDMerge (from the 1stWordPlus package). It worked on drive 0 but not when it was run from drive 1. **A**

Hints and Tips...

• **BASIC Editor.** If you type "EDIT." instead of just "EDIT", it gets you back into the editor at the same place you left it. This can be very useful when editing long programs. It even seems to work if you load and save other programs in the interim as long as you get the original program back into memory. Also, you can say EDIT Fred and it will take you to the first occurrence of the word "Fred" in the file. So, with judicious use of REM statements, you can jump into the program at just the right place every time. If you want to find DEFPROCfred, EDIT fred will probably bring you to PROCfred rather than the definition itself, but REM !FRED above the top of the procedure would allow you to type EDIT !FRED (note, all uppercase, which is easier to type) to drop you straight to the desired place in the program.

• **System DeltaPlus.** If you want to use two drives, issue the following two commands from the card index:

```
*dir :1
```

```
*set Delta$WPPrefix :0.$
```

Then don't specify the drive number in the data file name.

• **Artisan – fast load and save.** If you have typed in the Fastload/Save module from the March issue of Archive, *COPY FastRm onto the Artisan disk and then LOAD "ART3", list line 270 and add *RMLOAD FastRmlM and then re-save it. Now LOAD "ART5" change line 2540 to OSCLI"FastLoad "+f\$ and line 2620 to OSCLI"FastSave "+f\$. Re-save it and you now have a new, improved disc facility for Artisan.

• **305, 310, 410 or 440?** Following on from the hint in the May issue (page 7) it's all very well to know what the page size is, 8k or 32k, but to differentiate between a 305 and a 310 or between a 410 and a 440, you need to know how much memory is available. One way of doing this is to set up a local error trap and do a *Configure Spritesize 128. If the machine has 1 Mbyte or

less it will generate a "Configure parameter too big" error. If it still generates an error when you use *Configure Spritesize 64 then the software knows it is running in a 305. We have not had chance to check this on a 305 or a 410, so please will someone let us know if it does **not** work.

• **View 3 & Viewstore 1.1.** Various folk are still having problems with these. If so, try the following as suggested by Sean Kelly – thanks.

Viewstore 1.1. Locations &ABE2, &ABE3 and &ABE4 should all be set to the value &EA.

View 3. Set locations &A8D6, &A8D7 and &A8D8 to the value &EA. Also, there are four places where you get LDA#&82, JSR Osbyte (i.e. &A9, &82, &20, &F4, &FF). This is used to check for the second processor on the BBC. So to fool it, you have to change each of these to: LDX#&FF, LDY#&FF, NOP. You can do this by poking &A2, &FF, &A0, &FF, &EA into each of the sets of locations: &8834 to &8838, &88D8 to &88DC, &895C to &8960 and &AF8D to &AF91. (I checked these locations against the B3.0 version in the Master 128's and it bore no resemblance to it. Apparently it refers to the A3.0 version. Ed.)

• **Improved sound.** If you have the colour monitor that comes with the Archimedes, you can use the Audio Input at the back of the monitor (a phono socket) to take a signal from the Headphones output (3.5 mm stereo jack) from the Archimedes. Just use one of the channels or combine them with 100R resistors. (If you're not quite sure how to do this, we'll try to give more details next month.) This give a much better sound and provides you with a volume control.

Next two hints are from Peter Kiggins...

• **When using a SWI from BASIC** which either takes or returns a string, it is not necessary to use pointers to strings. The SYS command knows that a pointer has been returned by the

SWI and copies the relevant text into the string. For example, instead of

```
$TxtPtr%=Number$
SYS"OS_ReadUnsigned",FromBase%
,TextPtr% TO ,,Decimal%
```

you can use the more readable and easier

```
SYS"OS_ReadUnsigned",FromBase%
,Number$ TO ,,Decimal%
```

And to read the command line which started the program, all that's required is

```
SYS"OS_GetEnv" TO Command_line$
```

• **BASIC commands TWIN and TWINO** can be used without Acorn's (editor) TWIN. How? When TWIN is executed, BASIC first converts the current program from internal tokenised format to plain text format; then it calls TWIN with the location of the text as a command line string. So all that is required is a program called TWIN which reads the command line and saves the relevant chunk of memory:-

```
10 REM >twin
20 REM **fails on very short texts**
30 SYS "OS_GetEnv" TO command_line$
40 posat%=INSTR(command_line$,"@")
50 poscomma%=INSTR(command_line$,
, ",",posat%)
60 possemic%=INSTR(command_line$,
, ";",poscomma%)
70 start$="&"+MID$(command_line$,
,posat%+1,poscomma%-(posat%+1))
80 finish$="&"+MID$(command_line$,
,poscomma%+1,possemic%-(
poscomma%+1))
90 INPUT"Enter file name: " file$
100 OSCLI("save "+file$+" "+start$+
" "+finish$)
110 OSCLI("settype "+file$+" &FFF")
```

The point of all this is that the BASIC Editor can now be used to produce text files – for wordprocessing, compiling, etc.

• **Auto-Booting** – some of the commercial auto-boot programs don't take into account the filing system! This can be very annoying if you have a ROM podule and your system is set to start up in the RFS. The answer is (i) the software

developers should produce proper auto-boot programs; or (ii) put *CON.FILE.ADFS in auto-boot program!

• **Hard Disc Auto-Booting** – some readers are lucky enough to have hard discs. However, life with a hard disc is not all a bed of roses! Having copied your application programs to the hard disc, you run into problems with the auto-booting routines. Here is one of many possible solutions. Indeed this program could be used for those of us with floppies, who have more than one application on a disc.

```
10 REM >$.!BOOT
20 REM *****
30 REM * Multiple Auto-Booting *
40 REM * written by Adrian Look *
50 REM * 22nd June 1988 *
60 REM *****
80 SYS "OS_Byte",161,16 TO
, misc_flags
90 auto_boot=(misc_flags AND %10000)
100 file$="boot"
110 file=OPENIN(file$):CLOSE #file
120 IF file=0 THEN PROCmenu
130 file=OPENIN(file$)
140 INPUT #file,boot$
150 CLOSE #file
160 IF auto_boot THEN OSCLI("REMOVE "
+file$)
170 OSCLI("RUN "+file$)
180 END
190
200 DEFPROCmenu
210 REM *** your own code ***
230 REM should exit with file$
240 REM containing the filename
250 REM of the program to be run.
260 REM e.g. PRINT #file,filename$
270 ENDPROC
```

The program checks whether a file (file\$) is on the disk. If it is, the program reads the filename contained in file\$ and runs it! Otherwise, the program should create file\$ – as indicated.

This method also allows you to write library programs which will: create file\$ and then run the boot program. Thus by running these library files you can also initiate your application programs – without the need for a menu!

• **Quazer Fix.** If you are using Quazer 1.40, here is a cheat, (for those who like such things!) which allows you to keep going a bit longer. Firstly, *LOAD QuazCode 9000 and then rename Quazcode as, say, OldCode. Now if you want to set the number of lives, you need to change location &3D058. It should currently contain 2, so to change the number of lives to, say 10, put ?&3D058=10. Now if you want to avoid being killed when you collide, change the contents of location &3D90B from &E2 to &F2 by putting ?&3D90B=&F2. Finally, *SAVE QuazCode 9000 +36520.

• **Passwords** if you want them. **Quazer:** 2 Lizards, 3 Spiders, 4 Pedantic, 5 Analysis, 6 Larkin, 7 Company, 8 Manual, 9 Trading **Hoverbod** passwords: 2 Minervas, 3 SirBarry, 4 ZotyBlob, 5 Flumpies, 6 Squidgee, 7 Wobblies, 8 Posskett.

The following were sent in by Mike Harrison (of Watford Digitiser fame). He's also sent us various other bits of information which appear elsewhere in this issue. Many thanks Mike.

Writing Modules

(Ref. issue 5, page 45) SWIs in modules must always use the 'X' version of the SWI and, where appropriate, check for an error and exit, preserving the value of r0 returned by the SWI (i.e. don't restore the old value of R0 if it had been saved). e.g.

```
.a module routine
STMFd R13!,{R0-2,14} \or however
    many registers you need to save
SWI "XOS_SomethingOrOther"
STRVS R0,[R13] \overwrite old r0
    with error pointer
LDMVSFD R13!,{R0-2,PC}
\ the last 2 statements could
    alternatively have been :
\ ADDVS R13,R13,#4 : LDMVSFD R13!,
    {R1-2,PC}
```

For short routines, or where several SWIs are used, the following is neater..

```
.a short routine
STMFd R13!,{R0-5,14}
SWI "XOS_SomeSwi"
SWIVC "XOS_SomethingElse"
SWIVC "XOS_YetAnotherSwi"
STRVS R0,[R13]
LDMFD R13!,{R0-5,PC}
```

In this example, all code after first SWI should be conditional on V clear, so if any SWI gives an error, the code will 'drop through' to the end.

Remember that some SWIs may return errors you don't expect – e.g. OS_WriteC could return an error when output is *Spooled, setting the screen mode could give 'Bad Mode' etc. – some thought is required to ensure that module code will be reliable in all environments.

When setting up offsets in headers and command tables, it is much easier to specify them when using offset assembly (OPT 4/6/7) – set O% to the code buffer, and P% to 0. Offsets can then be included directly with EQUd (e.g. EQUd moduletitle). Warning – if you are still using Basic 1.00, there is a bug in ALIGN when using offset assembly. Version 1.02 is OK.

The following assembler macro is useful when building keyword tables in modules (it assumes the use of offset assembly, as described above):

```
DEF FNcommand(I%,A$)
[ OPTpass : EQUd A$ : EQUd 0 :
ALIGN
EQUd EVAL(A$)
EQUd I%
EQUd EVAL(A$+"syntax")
EQUd EVAL(A$+"help")
]:=pass
```

OPT FNcommand("NewCmd",N) will create a table entry, using the labels .NewCmd .NewCmdhelp and .NewCmdsyntax (N is the information word). Note that 'pass' is the name of the pass variable – change this if you use a different name.

Podule Manager/Clock Slowdown

(Ref. issue 7, page 44) The problem here is that SWIs are passed to modules with interrupts disabled, so if the SWI takes a significant amount of time to complete, the clock will slow down (the mouse pointer will also become sluggish). I can't see why the podule manager should need interrupts off, so it's probably an unintended 'feature' – SWI code which takes time to execute SHOULD enable interrupts, unless they have to be off. (The digitiser module only runs with interrupts off when absolutely

necessary!) When interrupts have been disabled for significant amounts of time, the seconds count of the real-time clock gets out of step with the minutes, which then don't roll over at 59 secs – this looks weird if you don't know about it!

System Devices

(Ref. issue 6, page 41) There is a very interesting feature, which I haven't seen documented, concerning device oriented filing systems. When you set the printer type using *FX5,n where n is greater than 2, the OS looks for a system variable called printertype\$n – n being the value used for FX5. This variable holds the name of the output file or device to which printer output should be sent. The Econet module, NetPrint, uses this to set printer type 4 to "netprint:". What happens is that when you do <ctrl-B>, a file is opened, with the name given in this system variable, and <ctrl-C> closes the file. While this would normally send output to a hardware device, it can also use the name of a normal file; e.g.

```
*set printertype$5 outfile
*fx5 5
```

Will cause subsequent printer output to be written to file "outfile" – handy for debugging, and saving data to print on other machines. Note that this will only work if a program does VDU2, sends ALL its output, then does VDU3. If it keeps turning the printer on and off all the time with VDU2/3, only the last part of the output will be in the file, as VDU2 re-opens the file, discarding its previous contents. (Econet users know only too well the problems caused by software not handling the printer correctly!)

Just for fun, try :

```
*set printertype$5 vdu:
*fx5 5
<Ctrl-B>
```

Vdu output will appear ttwwiiccee!! – once via the normal route, and once via the printer stream.

Use of SYS

(Ref. issue 9, page 47) The following is a bomb-proof version of OSCLI in BASIC, which will

report errors without entering BASICs error handler (it can also be used for other SWIs – e.g. OS_File to trap filing errors neatly)

```
DEFPROCoscli(OS$)
LOCAL flag%,error%
SYS"XOS_CLI",OS$ TO error% ; flag%
IF flag% AND 1 THEN P."Error : " ;:
    SYS"OS_Write0",error%+4
REM the error number can be read
    using !error%
ENDPROC
```

It works by reading back the V flag and error pointer returned by the X version of the SWI.

TWIN

(Ref. issue 7, page 7) You can warm start Twin using *Twin -warm, using the same version of Twin, of course – *GO may cause problems if the code has been corrupted, (or you've forgotten the address!). There is a bug in all the versions of Twin I've seen (I'm not sure what version has been released), which means that when exiting from TWIN to BASIC, the event vector is messed up, so if you overwrite the address where Twin was run, and enable events, the machine crashes. This can be avoided by doing QUIT, *BASIC, and OLD.

In one issue, there was a tip for entering ARMBE automatically at the line in error – here's a similar routine if you prefer using TWIN to edit BASIC programs – It uses the TWINO8 option to strip line numbers (which you don't need to use, DO YOU!!). If the error wasn't Escape, pressing <Y> will enter TWIN at the line where the error occurred. Note that it assumes lines are numbered in 10s – this will be the case if the program has been edited in TWIN without line numbers, but not if lines have been inserted/deleted from the BASIC prompt – in this case, enter TWIN, and exit again, or do RENUMBER.

```
ON ERROR PROCERR
...
...
DEFPROCERR
REPORT:PRINT" at line "ERL
IF ERR=17 END
PRINT"EDIT ?":IF GET$="Y" ELSE END
*FX138 0 129
```

Hints and Tips

```
*FX138 0 128
FORA%=1 TO LEN(STR$(ERL DIV10))
SYS 6,138,,ASC(MID$(STR$(ERL DIV10)
NEXT:*FX138 0 13
END
```

Hidden Software

(Ref. issue 9, page 4) Take a close look at the digitiser module!

• ARM ADR Macro – Adrian Look

The ADR command provided by the ARM assembler is essential for writing relocatable code (as all our ARM code should be?). If you assemble your code in absolute terms and it is executed at a different address it will fail because the addresses will all be wrong!

The ADR command overcomes this by calculating the difference between the address required and the program counter. It then assembles either an ADD or SUBtract instruction using the PC and the 'difference'. This means that addressing is relative to the PC. So wherever your code, it will always calculate the right addresses. For example:

```
.label
:
ADR R0,label
```

becomes:

```
SUB R0,PC,#(PC-label)
```

As you can see the SUB (or ADD) command uses an immediate constant, denoted by the #(PC-label). This constant has only 12 bits allocated. Eight of these hold a value and the other four indicate how much it is to be shifted i.e. %1111 1111 shifted by %1111 bits.

Knowing this we can see that we are able to use the ADR command to create a relative address of &AA00 i.e. %1010 1010 shifted by %10 bits, but &101 (%1 0000 0001) is impossible! If your address is just outside the reach of the ADR command, you can't use relative addressing.

Well, not unless you use another bit of code. The trick is to load the location to be addressed into a register and then perform the SUB (or ADD). I have done this in the procedure below.

However, this method uses more space and takes longer to execute. So where possible use the ADR command.

REM - the ADR macro -
REM reg% - register holding address
REM adr% - the address
REM opt% - the assembler option

```
DEFPROCadr(reg%,adr%)
IF reg%<0 OR reg%>14 THEN ERROR 1,
    "No such register in ADR macro"
IF (opt% AND 1)>0 PRINT'"->ADR macro"
reladr%=adr%-P%-20
sign%=SGN(reladr%)
reladr%=ABS(reladr%)
[OPT opt%:MOV reg%,#(reladr% AND &FF)
ADD reg%,reg%,#(reladr% AND &FF00)
ADD reg%,reg%,#(reladr% AND &FF0000):]
IF sign%<0 THEN
[OPT opt%:SUB reg%,PC,reg%:]
ELSE
[OPT opt%:ADD reg%,PC,reg%:]
ENDIF
IF (opt% AND 1)>0 THEN PRINT'"-> R";
    reg%;" contains "&";~adr%
ENDPROC
```

An here's an example of how it may be used:

```
10 REM >$.reljump
20 REM *****
30 REM *   ADR MACRO   *
40 REM *   EXAMPLE    *
50 REM *   Adrian Look *
60 REM *   3rd June 1988 *
70 REM *****
90 DIM test% &201
100 DIM string% &FF
120 $string%="Allo allo!" + CHR$(10)
130 FOR opt%=1 TO 3 STEP 3
140     P%=test%
150     PROCadr(1,string%)
160     [OPT opt%
170     .loop
180     LDRB R0,[R1],#1
190     SWI "OS_WriteC"
200     CMP R0,#13
210     BNE loop
220     MOV PC,R14
230 ]
240 NEXT opt%
250 PRINT'"Output of program: ";
260 CALL test%
270 END
280
290 DEFPROCadr etc as above
```

A

The Hardware Column

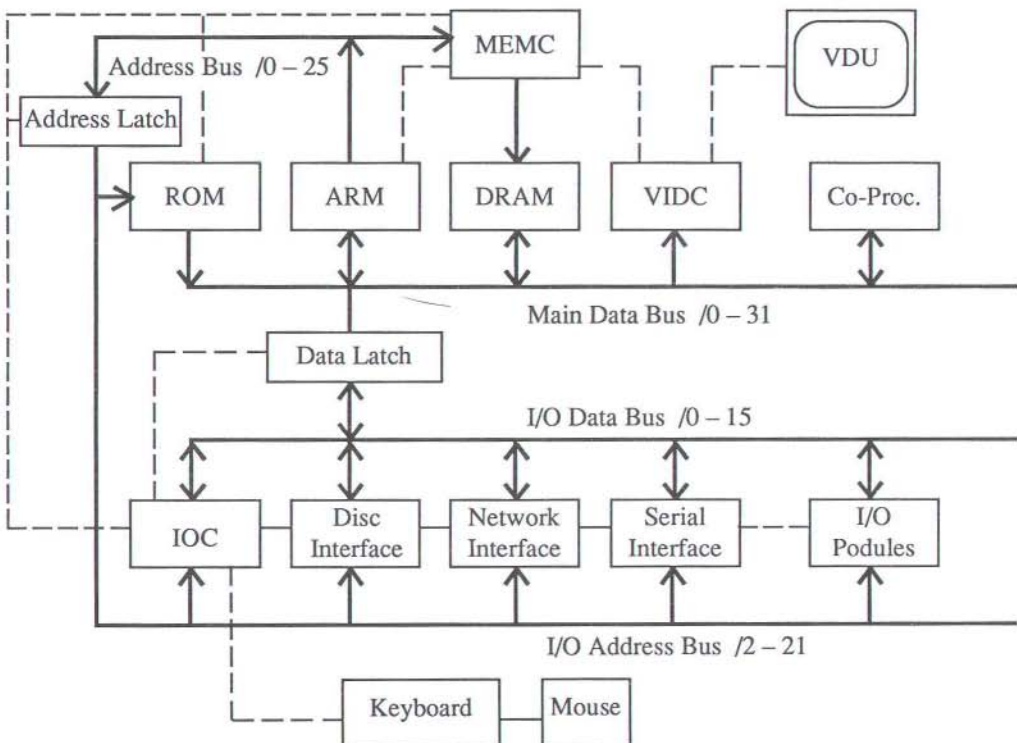
Brian Cowan

Overview of Archimedes Hardware

This is the first of a (hopefully) regular column on hardware aspects of the Archimedes. During the next few months we shall be taking a look at what is inside the box. There are many interesting and novel aspects of the Archimedes. As well as the RISC processor, the ARM, there are three other custom chips in the family. The memory management chip (MEMC) the video controller (VIDC) and the input/output controller (IOC) together with the ARM were designed to go together with the minimum of interfacing "glue" chips. In this way the chip count of the machine has been kept down together with the cost, but at the sacrifice of some generality of the individual custom chips.

Bus Latches

This month we start with a general overview of the system. A block diagram of the Archimedes is shown in the diagram. Note the address and data busses. The first thing to observe is that there are two of each. The system data bus and the I/O data bus are separated by a data latch and the system address bus and the I/O address bus are separated by an address latch, both latches being controlled by the IOC. The main reason for the latches is speed. The ARM is capable of operating at high speeds; the clock runs at 8 MHz although it is possible for it to run much faster. Speed of the system is limited by the speed of the RAM and, as the diagram indicates, ARM - RAM transactions via the MEMC take place unbuffered at the full system speed. Also VIDC operations occur at full speed.



Archimedes Block Diagram

All other transactions occur at slower speeds and thus operate through the buffers. The buffers hold their values allowing the ARM to continue to run at full speed while the peripherals give and take data asynchronously at their leisure. Note, however, that the system ROM is on the slow side of the latch. This reflects the fact that ROM operates somewhat slower than RAM; thus the provision of the faster RAM BASIC.

Data Bus Size

Another important point is the size of the data busses. The ARM is a 32 bit processor; this is the width of its data bus. The main data bus is therefore 32 bits, and RAM, ROM, VIDC and possible coprocessors all operate with 32 bit words. However the I/O data bus is only 16 bits wide, so that all external communication takes place with data chunks of this size. The way the data latch operates is important in this context. Data is read into the lower 16 bits of the bus while data writes take place from the upper half of the bus.

Coprocessor Bus

It is now possible to see the difference between the implementation of coprocessor podules and other podules. The coprocessor podule requires access to the full 32 bit data bus while all other podules use only 16 bits. The 300 series machines do not provide the 32 bit main data bus on the podule socket, but only the latched 16 bit bus thus precluding the use of coprocessors. However, if access to the 32 bit bus can be made in some other way such as from the ARM or VIDC socket, a coprocessor could be attached.

Upgrades

We shall look at the question of RAM size in different machines in a future article. The conclusion, however, concerning both RAM and coprocessors, will be that the 300 machines can be fitted with simple add-ons to make them equivalent to the 400 machines!

Hints and Tips

Printer leads

It is a relatively simple job to make a lead to connect an Archimedes to a printer using ribbon cable and IDC connectors. To plug into the Archimedes, you need a 25 way "D" plug such as RS no. 472-269, and

for the printer (Epson types) you need a 36 way Amphenol type plug such as RS no. 470-954. These may be joined by a length of 25 way ribbon cable. The plugs are attached to the cable by carefully squeezing the plug back onto the ribbon. Be careful to arrange that pin no. 1 of both plugs connect together via the marked side of the cable.

If you already have a cable for the old BBC or Master then you can add on a 25 way "D" plug close to the BBC plug. You can then use the lead on either machine. To attach the Archimedes plug you will need to remove one wire from the ribbon cable at the plug. It is wire number 26 that must be cut away; be sure to leave number 1 intact. **A**

Small Ad's

If you would like to insert small ad's free of charge, send them to us. Each should be less than 30 (thirty) words long and should relate to Archimedes and associated devices, i.e. not adverts for secondhand BBC's and Masters!

- CUMANA CS400 D/Drive with PSU, 1 year old £85. Ring 0238 528422 after 6.00 p.m.
- Smith Corona D100 Printer, 1 year old, very little used. £90. Ring 0238 528422 after 6 p.m.
- A305 + colour monitor for sale with printer, modem, software, 1 meg upgrade. £850 o.n.o. Phone Peter on 01-727-5466. (Peter Lancaster, 74 Lancaster Road, London, W11 1QR.)
- A310M for sale with colour monitor & Prog Ref Manuals. £950 o.n.o. Phone (0272) 325040.
- Disc Upgrade kit (owner now has hard disc so doesn't need a second 3.5" drive) almost unused. £120 inc. fitting or delivery. Phone Whitstable (0227)-273796.
- GraphicWriter, sell for £13.50 (I now have 1stWPlus) and Arctist (Fairhurst Insts) sell for £9.50 (I now have Artisan). Phone Philip Hirons, 01-310-7162 (H) or 01-526-1578 (W).

MS-DOS Column

Ken Biddle

I would like to begin this month with a brief review of a fully buffered 5.25" disk drive interface produced by Dudley Micro Services for the Archimedes and explain how this can be used in conjunction with the P.C. Emulator.

The unit is very small and compact and comes with comprehensive fitting instructions. The interface is encased in a smart plastic case and fits very neatly over the top of the internal disk drive within the Archimedes. There is a trailing ribbon cable which then needs to be dangled out of the back of the case. Some people may not find this neat but one advantage over some other interfaces is that it does not use any of the back plane slots so they may still be used for other pieces of hardware.

The unit has a small pin switch on it so that the drive may be enabled or disabled from detecting if a disk is present or not. Once fitted, which really was simplicity itself, (even for me I and I do not often dabble inside the machine) you merely plug in your 5.25" drive into the female socket at the back of your machine and off you go.

One thing you must remember to do (which typically I forgot!) if you have double 5.25" drives is reset the drive allocation switches inside the external drives. This needs to be done because the internal Archimedes drive is configured as drive zero and one of your external drives will also be set as drive zero. You can accomplish this operation with remarkable ease – just take the lid of your drives and you should find the drive allocation switches labelled – 0 1 2 3 : if you move the pin setting on the drive currently set on zero to 2 then all should now be well. Incidentally PLEASE remember to carry out all of these operations with the computer and disk drives totally disconnected from the mains supply.

I telephoned Dudley Micro Services when I had a bit of trouble and they could not have been more friendly and helpful. If only more companies were like this then life would be much more pleasant.

Conclusion

I have been using the interface now for several months and have been very pleased with its faultless

operation. It is reliable, well made and, at £27.95, good value for money. The 5.25 inch disk interface can be obtained from: Dudley Micro Services 30 Hadley Close, Netherton. Dudley, DY2 9JX.

The sort of use that I have put it to is both transferring software from my BBC Master and also using it extensively with the P.C. Emulator and I have experienced no problems at all. Devices like this are of particular use with the P.C. Emulator since most MSDOS software still comes in 5.25" disk format.

Software that Works

From now on, with regard to software that seems to run O.K. on the emulator, I will try to include the version number of the emulator that it was tested with. So please when writing in to let me know of software that does run, say which version of the emulator you are using.

Equally, if anyone has any of the different versions of the emulator that we could have then we would appreciate them so that the column could give more accurate information relating to all versions.

The new updated software list that all appear to run with no problems on the emulator are as follows :

P.C. Emulator version 1.00

Wordstar Release 3; Wordstar Release 4 (used to write this article); Microsoft QuickBasic Release 3; Norton Utilities; Dbase III; Zortech C Release 2; Zortech windows (window utilities for Zortech C); Sidekick Release 1.56A; MSDOS Release 3.3; Multimate version 3.0; Lotus 123 (No version number specified); PCtools; Procomm; Symphony (No version number specified); Framework (No version number specified); Laplink; Sideways; Hunt for red October; Sherlock; Microsoft Word ver 2 and 4 (with the mouse); Cardbox Plus; Turbo C version 1.00; Turbo Pascal versions 3 and 4.

P.C. Emulator version 1.09

Turbo Basic version 1.0; Turbo Pascal version 3.01A; Turbo Pascal Graphics Toolbox; M TEC BBC BASIC; LPA Micro-Prolog Professional version 1.2; LISP/80 version 1.0; BASIC version 2.11; Printmaster; P.C. Tools version 3.1; Multiplan version 3.0; Dbase III +; Fontrix; Volkswriter Deluxe; Zortech C Release 2; Zortech Windows

(Window utils for Zortech C); Wordperfect versn 4. Public domain software: Trivia machine; Still river shell; P.C. Deskteam; P.C. Calc; P.C. Outline; Wagner file Utility; P.C. Write; Lady Bug.

P.C. Emulator version 1.2

I am afraid I do not have this version myself so I cannot test any of software that I have access too and nobody has sent me details of software running under the 1.2 version although it is presumably more compatible than the other versions.

I will use up what space I have left on the column to try and answer some questions that have been sent in. Due to space limitations I will not repeat the questions here but just make a series of statements in a hints and tips type fashion.

Ramdrive Problems?

I have had one or two comments about not being able to get the ramdrive working when I explained this in a previous article. All of the tips and or comments that I write about are tested first on the Archimedes using the P.C. Emulator. I am aware that there are several different versions of the Emulator software (I use 1.00) which may be different from the one that you may be using. This may obviously cause certain things to work differently.

External Disk Drive format

I have been asked in a number letters about the external disk drive always behaving as a 720 kbyte drive. The default configuration of the P.C. Emulation is that two drives are defined – A: and B:. If you add a 5.25" drive then it will be recognised as drive B: with 80 tracks, 9 sectors per track, and 2 heads. This will give it a storage capacity of 720 kbytes just like the Archimedes internal drive.

This may be changed by using the line editor (EDLIN) and adding the following statement to the CONFIG.SYS file twice:

```
device=driver.sys /d:1 /t:40 /f:0
```

by entering the statement twice it enables you to copy from disk to disk on the same 5.25" drive.

You should now re-boot the system (ctrl/alt+delete) and the system will configure two new external drives. (The system will tell you what the drive letters will be.) On my system the Drive letters are D: and E: The drives will be treated by MSDOS as 40 track drives of 360 kbyte capacity.

Not only can you access the drive as 360 kbytes but you may still use it as a 720 kbyte drive if you wish.

If the drive letters are as I have quoted above then you simply decide which capacity drive you wish to use by accessing it with the appropriate drive letter, for example :

FORMAT B: – Format external drive to 720k.

FORMAT D: – Format external drive to 360k.

You must remember that if you do use the external drive with a 360k format to switch the 40/80 track switch to the 40 track position.

- **P.C. Emulator vs Master 512.** I have received several letters relating to software running on the Master 512. You must remember that the Master 512 does not run MSDOS but DOS Plus which is a hybrid of CPM86 and DOS. Although many MSDOS programs will run on the 512 the Archimedes P.C. emulator is a much more compatible environment. The one advantage the Master 512 has is that it is much faster.

- **Ramdrives.** When a Ramdrive is set up (by amending the CONFIG.SYS file – see Archive Volume 1 No 8 , Page 23) the system will display on the screen while booting which drive letter you will need to use to access it. You may be interested to know that more than one ramdrive may be set up merely by duplicating the inserted statement in the CONFIG.SYS file.

- **Using the mouse.** I have had several requests about software using the mouse. Many people cannot get the mouse to work at all or in some cases only partly. I am afraid I personally do not have any software that uses a mouse but I assume the problem lies in the fact that the mouse system driver that you are using just does not work under the version of the emulator that you are using. If anyone has any software running OK with the mouse or can provide any information then please let me know so I can pass it on.

Well folks, that's all for this month. Don't forget to write in with Hints, Tips, and/or Queries – the more people that write in, the better we should be able to make the column reflect what you want to read. It doesn't matter how trivial you think it is, someone out there is probably begging for the answer to a problem that you may think is obvious. **A**

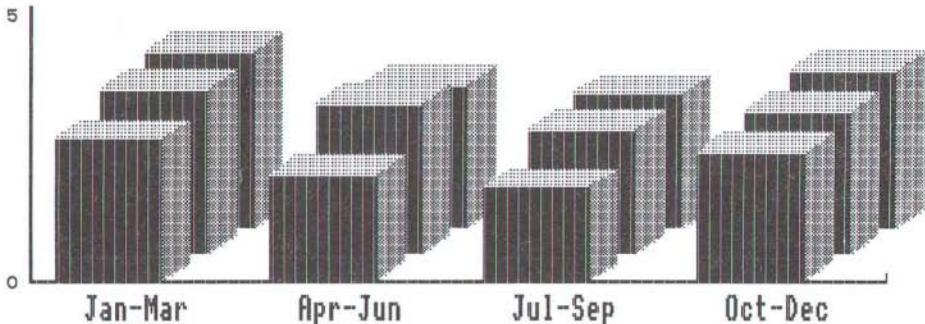
PRESENTER

Affordable Presentation

Graphics for the

Archimedes

PRESENTER is a truly professional graphics program which allows the user to create, modify, print out or photograph high quality colour displays of data in either bar pie or line format. Data is entered either manually in spreadsheet fashion or can be imported from other spreadsheets or programs in comma separated format. **PRESENTER** makes full use of the **Archimedes**' WIMP environment and high resolution graphic modes and incorporates 3D displays, auto-scaling axis, user definable layouts and multi-layer graphs. Screens may be saved for use with graphic packages like **ARTISAN** or graphic wordprocessors like **1ST WORD PLUS** & **GRAPHIC WRITER**. **PRESENTER** can also be used with **PIPEDREAM**.



At only £24.95 ex-VAT, **PRESENTER** is an invaluable extension to existing spreadsheets and wordprocessors. It is a **must** for all Archimedes' users.

PRESENTER is available from all good Acorn dealers or direct from LINGENUITY.

```

Please send me...PRESENTER Packs      Retail Price    £24.95   £.....
                                          VAT            £ 3.74   £.....
                                          Postage UK     £ 1.15   £.....
                                          Postage Overseas £ 6.00   £.....

                                                    TOTAL    £.....

Payment   Visa...   Access...   Cheque...

Credit card number [ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ]
Card Expiry Date... / ...
Name(as on credit card) .....
Address .....
                ....Postcode .....

Send to LINGENUITY, Dept.JT, P.O.Box 10, Halesworth, Suffolk IP19 ODX.
```

Telephone orders accepted - ring 098 685 476

BASIC V Utilities Forum

Clifford Hoggarth

This section of Archive gives you another opportunity to contribute. If you have written any useful routines which may be of interest to others, send them in to the editor! If you think that you could write (or have written) a better routine than one already published, send it in!

Alternatively, have you got a problem? (Who hasn't?! – but we're talking about problems with BASIC.) Unsure of how to write a particular routine? Send in a description of what you need and we will ask for people's ideas. (For a fast response, bear in mind that the magazine goes to the printers around the 22nd of each month – missing that deadline means an extra wait.)

Hopefully, through this column we should be seeing routines refined as time goes by until we have the neatest, fastest routines for performing various particular tasks.

Here are a couple of 'quickies' to be going on with....

Easier and Faster SYS's

When using SYS calls, as described in last month's issue, it is quicker to call the routines by number than by name because the saves the overhead of Arthur having to look up the number. The disadvantage is that using the number makes a program much less readable. This can be overcome by assigning a variable with the corresponding number. For example,

```
OS_Byte=6
SYS OS_Byte,15,0
```

Using a variable is also slightly faster than using a constant and it means you don't have to use some of the extremely long SWI names, thus

```
SYS "OS_ConvertStandardDate
      AndTime"
```

could be replaced by:

```
Date=&C0
SYS Date
```

which is a lot easier to type if it is needed repeatedly!

Which way to go?

When a choice has to be made between zero, a negative value and a positive value, e.g. the movement of the mouse from its previous position, the choice of action can be made using the CASE statement and the SGN of the value, thus:

```
CASE SGN(displacement) OF
WHEN -1 : PROCnegative
WHEN 0  : PROCno_change
WHEN 1  : PROCpositive
ENDCASE
```

This also has the benefit of preserving the value of the variable, displacement. **A**

DESKTOP GAMES

FOR ARCHIMEDES 310 410 440

A suite of FIVE programs for extending the desktop on the ARCHIMEDES 310, 410 and 440 computers... comprising of THREE logic games and TWO utilities.

- CRACKCODE** – Can you break Archie's colour code in ten moves?
- LEAP FROG** – Three different versions of Solitaire.
- SLIDERS** – Sliding block puzzle with two grid sizes and complete with FIVE pictures to solve. Also contains a utility to convert your own pictures into puzzles!
- OS COMMAND** – Access the operating system from within the desktop.
- VOLUME** – A sliding volume control for Archie.

"It's nice to see good quality software being sold at a sensible price... Highly recommended" – Archive magazine, May 1988.

Also contains a HIDEAWAY 100% ARM-code version of Invaders. Can you find it...?

Send orders to:-
GEM ELECTRONICS,
17, Tandragee Road,
Portadown,
Craigavon. BT62 3BQ



AND ONLY £5.95 inc.

Overseas please add £1.00

REAL GEMS AT SPARKLING PRICES!

Leonardo

Mode 12 Archimedes Art Package

Designed around a pixel editor but with a comprehensive set of drawing options available

at all levels of magnification.

**£17.50
only**

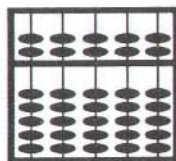
Available from: **BEARD TECHNOLOGY**

111 EVERING ROAD LONDON N16 7SL

**£17.50
only**

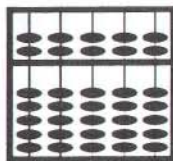
Floating Point Assembler

- Use FP mnemonics from within the BASIC assembler
- FP mnemonics include division, logs, trig functions, square roots
- Extended Precision gives 19 significant figures (base 10)
- Conversions between Scientific Notations & Packed Binary Coded Decimal
- Written in BASIC for use with APPEND, LIBRARY & INSTALL
- Accepts BASIC variable names for FP registers & immediate values
- Program examples include extended precision square roots and pi



ABACUS TRAINING

29 Okus Grove, Upper Stratton,
Swindon, Wiltshire. SN2 6QA
Tel: (0793) 723347



Price: £12.50 including VAT and UK postage

First Word Plus Review

Mark Sealey

One of the very first things you are likely to want to do with your Archimedes is use it for word-processing: the memory is there for long documents, the speed for checking spellings and the screen handling for fonts and appearance. At the moment, there are really three contenders and although this review is not a comparative one, readers may want to look at Archive number 6 (page 12 for 'Graphic Writer' from Clares, and page 10 for a brief look at the diminutive 'ArcWriter') as well as number 9 (page 4 for the latest on free ArcWriters or half-price First Word Pluses). You may even be so fond of Wordwise or View that you will want to run them under the emulator. In this case various issues of Archive have dealt with these. Not to mention PC packages like Word-Perfect, Word and Wordstar....

No, what is needed here is a cool, hard look at a product costing the better part of £100 (only £20 less than the regrettably high standard Acorn Archimedes price) for something that ought to be really special. Something that takes advantage of those Archimedes features mentioned just now. Is First Word Plus, originally written for the Atari, worth all that money? And what will you get in the package anyway?

Features

To answer the second question first, you get a WYSIWYG word-processor which is very user-friendly and can incorporate simple graphics, a mailmerge suite, umpteen printer drivers with a customisation routine, a 40,000 word dictionary with the option of continuous checking and of adding items of your own, sample texts, an excellent manual... and a buzz. At least on machines that I've heard or heard of. Unless you have got the fix, the continuous buzz is as bad as on almost anything around - except 'Arcwriter'.

There, though, the comparison with the other Acorn offering (or with 'Graphic Writer', for that matter) ends: First Word Plus far outstrips anything that has ever been available for BBC machines until now and is streets ahead of Word and Wordstar in facilities and ease of use as well.

Many word-processors use the idea of toggling between a 'Command' and 'Text' mode (usually by pressing Escape). This one, being entirely WIMP driven, is much more sophisticated. Indeed, if you are not well up in the hierarchical structure of ADFS, you might find locating and renaming some of your files, as well as preparing a working disk, quite daunting. Several times the manual just says ".....refer to your Archimedes User Guide". The components of the Suite are spread over some 12 directories!

Performance

The idea of using different directories is a good one but this is the first place where you get the distinct impression that release of the product was hurried by Acorn: there is no documented way, for instance, of allowing files onto Drive 1 by default, though in fact it is quite easy. Other signs are the absence of a swap-case function or delete forward (e.g. Delete up to Character, as in View), for instance. Delete to end of line can be achieved by <ctrl-copy> and delete word right by <ctrl-copy>, though these are not documented. Acorn are aware of some of these shortcomings and a future version of the package may put this (and other points raised here) right.

Ease of Use

Text is typed onto a white Mode 12 screen inside a Window, the size and layout of which can be controlled in various ways. By conventional WIMP practice (scroll bar etc), by one of the many pop-up menus and by setting the ruler to control what comes underneath it.

This multiplicity of options is very intelligently handled by the program: one of the windows (all of which are models of clarity) controlling page-layout, for instance, updates the expected default page width as you select elite, pica or condensed styles etc. You can see ALL effects (those, as well as italic, underline, non-European and the rest) on screen as you type. First Word Plus is full of touches like this. And others: deleting a TAB, for instance, has the effect of smoothly sliding text left to its new position. The facilities for indenting are legion and work very well. No more positioning separate rulers

to do it all paragraph by paragraph. Then the footnote commands; these make the sort of work (like academic texts) that needs them, a real joy.

There is just the right amount of dialogue in the windows and each previous version is automatically saved into a 'bak.' directory when you save the current one. There is a 'save and resume' option, extra on-screen help, up to four documents to be open simultaneously with facilities for moving and copying text between them. You quickly become very familiar with this system, which after all is designed to wring the most out of the machine: options not possible at any particular time, for instance, are shown in grey and cannot be clicked on. (*A familiar idea to a Mac user! Ed.*)

Documentation

As stated earlier, the manual is outstanding, well indexed and takes both a tutorial and a reference approach, though there are omissions. The sections on First Mail are as good as the rest of the 250 pages and the parts that deal with overall layout are more than adequate if you are new to word-processing. In that sense, it is an ideal package in that it is both easy to get up and running in moments and yet offers a wealth of facilities, which you will enjoy using.

Mail-Merge and Dictionary

You can create data and command files to handle who gets what in the case of a duplicate letter or circular. There is no mention, though, of using data from other Archimedes databases such as System Delta Plus or Alphabase. That would have been useful here and is surely essential when you think that this package will be competing, say, with those running under Microsoft Windows and other integrated environments. Money to be made by someone writing utilities to do this.

Where the dictionary is concerned, the speed is good and you can add multiple auxiliary word-files of your own. On the whole, business and professional users will be no less pleased with First Word Plus than hobbyists and general-purpose users.

Conclusions

There are many small blemishes: proportional spacing isn't possible (*'tis now - see page 46. Ed.*), the graphics can't be scaled (though they can be moved) on screen, the date from Arthur can't be incorporated, there is no Import or Export of files

other than First Word Plus ones. You can't get to the end or beginning of a document with a single key-press (like ctrl + up-arrow). The identification of blocks and text marked as italic, bold etc didn't work smoothly. All these detract from an otherwise excellent word-processor and - unless they are put right - may mean it won't hold the field for long, even though at the moment it is definitely the best buy. **Recommended, with reservations. A**

If you bought your Archimedes before 1st April, you can get 1stWP at half price from Acorn (£45.97), otherwise, you can buy it from Archive for £85.

Colour Palette for First Word Plus

Apologies to the author of this contribution. His letter got separated from the pages of text and I couldn't remember who sent it in. Moral - put your name on everything you send in - including program discs, please.

Although first impressions of the new Archimedes word processor, First Word Plus, are good. There was one point which definitely did not meet with my approval; the colours chosen for the display. This is because the background is white or very pale coloured with the result that my eyes are troubled by the apparent flashing of the display.

Exploration of the manual did not reveal any information on changing the colours. However, investigation of the disc showed that a file *!palette* was included. This file is compatible with the Desktop module and can be altered by it to suit your preferences. The only difficult point is deciding which colour is used for which function on the First Word Plus displays.

The following information gives the results of my investigations together with a suggested colour scheme for those who like a black background for the main text area. The colour numbers are the numbers of the colour boxes on the Desktop palette editor starting from 0 at the top left and going from left to right and top to bottom to give colour 15 at the bottom right corner as in the following diagram.

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

No	Colour	Use
0	Black	Background
1	Red	Left hand bar of main text window, Menu box background
2	Green	Light text, File name caret, Menu items not available, Base screen background
3	Blue	Marked normal text
4	Blue/green	Normal text
5	Purple	Marked light text
6	Cyan	(Not used)
7	White	Highlighted text background, Bold text, selected menu items, most other text

8	Light Grey	File selection box border
9	Beige	Menu box border
10	Brown	Default confirmation box background
11	Yellow	Title bar background
12	Blue	Scroll bar
13	Red	Title bar foreground (text)
14	Pink	Scroll bar background, Menu text (unselected items)
15	White	Menu text (selected items)
Ms1	White	Mouse pointer border
Ms2	Blue	Mouse pointer interior
Ms3	Beige	Hour glass sand colour used for the busy icon.

It is hoped that this information will be of help in enabling you to set up your own colour palette. It certainly will give you many hours of pleasure in setting the colours exactly to your liking! **A**

Auto Dating 1st Word Plus Documents

Tim Powys-Lybbe

This program is to be used with 1st Word Plus on the Archimedes. Its purpose is to put today's date in a file called 'format'. To use it, you must first prepare a file in the 'doc' directory called LetterBlnk. This file should contain your address, any logo you want and the date in this form:

Wednesday 31st September 1999

The reason for using Wednesday and September is that these are the longest names of days and months respectively.

The program will change this date to today's date, in the same format. The resulting file will then be saved in the 'doc' directory with the name 'format'.

When you create any new file in 1st Word Plus, it calls up any file with the name 'format' and loads it in at the start... The resulting date is fixed and will not change if you want to reprint the letter at a later date (a fault of the "letterdate" feature within 1st Mail Plus).

You should put this program in your Library directory on the 1st Word Plus !BOOT disc and also amend the !BOOT program to include this command:

```
*key 1
```

```
QUIT|MLetterdate|M1stWord+|M
```

This will update the 'format' file each time you start up 1st Word Plus to the current date.

(Tim has also provided us with a printer configuration file for the Kaga Taxan 810 which, he says, should also work for the Kaga 910 and the Canon PW-1080. These are on the monthly program disc.)

```
10 REM > $.1ST_WP.LetterDate
30 REM Tim Powys-Lybbe
50 DIM O%50
70 REM First find the length of
   LetterBlnk and put in L%
80 Source$="$.1wp.doc.LetterBlnk"
90 PROCOfsFile(Source$)
100 IF A%=0 PRINT'"Cannot find the
   file: "Source$": END
110
120 REM Load the LetterBlnk file and
   find where 'Wednesday' is.
130 L%=E%
140 DIM File%L%
150 OSCLI("LOAD "+Source$+" "+STR$~
   File%)
```



```

160 TheDay$="Wednesday"
170 Day$=""
180 FOR I%=0 TO L%
190   A%=File%?I%
200   Day$=Day$+CHR$(A%) ELSE Day$=""
210   IF INSTR(TheDay$,Day$)=0
       Day$=""
220 IF NOT (Day$=TheDay$) : NEXT
230 IF I%>=L% : PRINT "Cannot find
    'TheDay$' in the file called
        'Source$'..." : END
240
250 REM Save where 'Wednesday' was
260 Date%=I%-LEN(Day$)+1
270 I%=L% : NEXT : REM Tidy up job!
280
290 REM Get current date and convert
    to the desired format
300 DIM T% 25, D% 50
310 ?T%=3
320 SYS "OS_Word" ,14,T%
330 S%=100 : DIMB%S%
340 Design$="%WE%0%ZDY%ST%0%MO%0%CE%
    YR%0"+CHR$(0)
350 $D%=Design$
360 SYS "OS_ConvertDateAndTime" ,T%,
    B%,S%,D% TO X%,Y%,Z%
370 K%=X%
380 Date$=""
390 FOR I%=X% TO Y%
400   IF ?I%=0 ?I%=13:IFI%<Y% Date$=
       Date$+$K%+" " :K%=I%+1
410 NEXT
420
430 REM Replace old date with current
    date
440 I%=Date%
450 REPEAT I%=I%+1
460 UNTIL File%?I%=&A
470 DateEnd%=I%-1
480 $(File%+Date%)=LEFT$(Date$,LEN
    (Date$)-1)
490 $(File%+Date%+LEN(Date$)-1)=
    STRING$(DateEnd%-Date%-LEN
    (Date$)+1,CHR$(&1E))
500 ?(File%+DateEnd%)=&A
510
520 REM Save file as 'format'
530 Output$="$$.lwp.doc.format"
540 PROCosFile(Output$):IFA% OSCLI
    ("ACCESS"+Output$+" WR")
550 OSCLI("SAVE "+Output$+" "+STR$~
    File%+" "+STR$~L%)
560 GOTO650

580 DEF PROCosFile(A$):REM Checks if
    the file A$ exists.
590 $O%=A$
600 R%=5
610 SYS "OS_File" ,R%,O% TO A%,B%,
    C%,D%,E%
620 ENDPROC
630
640 REM Debugging routines.
650 *K.0 F.I%=File%TOFile%+L%:P%=
    ?I%:IFP%>31 ANDP%<127 P.CHR$
    (P%) " ";:N.:P. ELSEN.:P.|M
660 *K.1 FOR I%=Date% TO Date%+100:
    P.;?I% " ";:N.|M
670 *K.2 OSCLI("DEL."+Output$)|M
680 *K.3 OSCLI("DUMP "+Output$)|M
690 *K.4 OSCLI("DUMP "+Source$)|M A

```

1st Word Plus Notes

Brian Carroll

Although I had been planning to wait for the Computer Concepts ROM word processor, I succumbed to Acorn's half-price offer of 1st Word Plus instead of Arcwriter. I quite like it on first acquaintance; it has some very nice features, but a few surprising omissions and problem areas as well.

Transferring Files

Because of the way it re-formats and pads spaces it is difficult to transfer ASCII files directly from other word processors such as Wordwise Plus. Spooled output from the latter ends lines with a plain <CR> and spaces are 'hard' (&20), but 1st Word Plus needs <LF>'s not <CR>'s and will not reformat paragraphs with hard spaces. I have written a short utility (which Adrian Look has converted to machine code for speed and added one or two extra features such as tabs and pound signs) to convert to a format which can be read in and converted into a proper 1st Word Plus document. Load the converted text into 1st Word Plus, select "WPmode on" from the edit menu and then reformat the whole text by using "Style" to "reformat" to "whole text".

Headers & Footers

The ruler controlling these can only be inserted once, as the very first line of the text area, and it governs both, so it is not possible to have different pitches top and bottom or more than one text line.

Control of Character Pitch

This is controlled by the preceding ruler, so although style may be altered word-by-word, pitch may not. Such a facility is one of the major blessings of Locoscript!

Printer Drivers

A good range of drivers is supplied, but as the manual thoroughly emphasises, 'Epson compatible' does not necessarily mean 100%; this applies to my Canon PW-1080A. With the aid of the excellent printer driver Appendix, I have made drivers for the Canon for single-sheet and continuous paper. They should work for the Taxan 810 and the Canon PW-1156 as well. (*N.B. single-sheet drivers do not work due to a bug which Acorn are aware of – the printing does not stop at the end of each page! Ed.*) The 'source' and 'object' files are on the program disc in directories 'hex' and 'cfg' respectively. There is only one snag: Canon NLQ works only in 10 chars/inch, so 'expanded' NLQ is 5 chars/inch not 6 as 1st Word Plus inflexibly assumes. One cannot therefore centre expanded NLQ headings correctly, but must resort to counting characters and padding with spaces!

ASCII Output

Conversely, ASCII output is easy: it is simply necessary to select WP mode OFF from the edit menu and save the file (under a new name). But you cannot keep the formatted version, e.g., justified, and another snag is that the file contains no <CR>'s, just <LF>'s, so there would be problems with some uses of the ASCII version. **A**

```

10 REM >$.ConvertWP
30 REM *****
40 REM *   Converting WwPlus   *
50 REM *   files for 1st WPlus  *
60 REM *   written by Adrian Look *
70 REM *   24th June 1988      *
80 REM *****
100 special_char=&FF
110 line_length=80
120 new_line=10
130 MODE 0
140 PROCInputs
150 PROCset_up_code
160 PROCconvert
170 END
180
```

```

190 DEFPROCInputs
200 INPUT "Source filename: "s_file$
210 file=OPENIN(s_file$)
220 size=EXT #file
230 CLOSE #file
240 INPUT "Destination filename: "
                                     d_file$
250 ENDPROC
260
270 DEFPROCset_up_code
280 DIM sorc% size
290 DIM dest% size*1.3 :REM a bit
                                     bigger than souce file
300 DIM data% &100
310 DIM code% &100
320 PROCAssemble
330 offset=0
340 REPEAT
350   READ a$,b$
360   IF a$<>"****" THEN
370     ?(data%+offset)=EVAL(a$):
                                     offset+=1
380     ?(data%+offset)=EVAL(b$):
                                     offset+=1
390   ENDIF
400 UNTIL a$="****"
410 ENDPROC
420
430 DEFPROCconvert
440 OSCLI("LOAD "+s_file$+" "+STR$~
                                     (sorc%))
450 C%=sorc%-1
460 D%=dest%
470 E%=data%
480 F%=sorc%+size
490 G%=data%+offset
500 CALL convert
510 end=!data%
520 OSCLI("SAVE "+d_file$+" "+STR$~
                                     (dest%)+ " "+STR$~(end))
530 OSCLI("SETTYPE "+d_file$+" FFF")
540 ENDPROC
550
560 DEFPROCAssemble
570 FOR opt=0 TO 3 STEP 2
580   P%=code%
590   [OPT opt
600   .convert
610   STMPD R13!,{R0-R12,R14}
620   .start
630   LDRB R0,{R2},#1
640   ADD R9,R9,#1
650   MOV R7,R4
660   MOV R8,#0
```


More Mode Conversions

Robin Newman

Robin has sent us two BASIC programs and two machine code utilities which will convert mode 7 screens either to mode 4 or mode 9. He has also improved on Mike Williams' program from last month allowing it to convert mode 13 to 15 correctly, taking into account any defined palette. The listings were far too long, so we've put them on the monthly program disc, but here are the mods...

These modifications to Mike Williams' program (June, page 43) first of all read the colour definitions from the source screen file into a byte array, pal%. The size of the array is calculated from the offset information pointing to the start of the screen image in bytes &30 and &31 of the screenfile. (lines 161 to 166) This array holds colour information in a form suitable for use with the VDU19 command. For a mode 13 screen, there are 64 colours defined (colour numbers 0 to 63) and the command VDU 23, <column>, p, r, g, b defines the colour - p is &10 or &11 depending on whether the colour is 'flashing' and thus has one or two components. For non-flashing colours, the colour information is repeated twice with p=&10. Lines 1111 to 1117 read the stored array and define the colours for the new mode. Note, 1114 and 1115 are identical! Line 170 can be deleted as the screen pointer is now already in the correct position.

```

670 .loop
680 LDRB R1,[R7]
690 CMP R0,R1
700 BNE jump1
710 MOV R8,#1
720 LDRB R1,[R7,#1]
730 CMP R1,#special_char
740 BNE jump
750 CMP R9,#line_length
760 BMI jump1
770 MOV R1,#new_line
780 .jump
790 STRB R1,[R3],#1
800 CMP R1,#new_line
810 BNE jump1
820 MOV R9,#0
830 .jump1
840 ADD R7,R7,#2
850 CMP R7,R6
860 BNE loop
870 .skip
880 CMP R8,#1
890 BEQ cont
900 STRB R0,[R3],#1
910 .cont
920 CMP R2,R5
930 BNE start
940 STR R3,[R4]
950 LDMFD R13!,{R0-R12,PC}
960 ]
970 NEXT opt
980 ENDPROC
990
1000 REM ***bytes to be changed***
1010
1020 DATA &20,&1E
1030 DATA &20,&FF:REM counter used
                        by program
1040
1050 DATA &0D,&0A : REM CR > LF
1060
1070 DATA &DD,&20 : REM TAB into
1080 DATA &DD,&20 : REM 5 fixed
1090 DATA &DD,&20 : REM spaces.
1100 DATA &DD,&20
1110 DATA &DD,&20
1120
1130 DATA 35,&A3 : REM # > £
1140 DATA 96,&A3 : REM £ > £
1150
1160 DATA &02,0 : REM <gr> to 0
1170 DATA &07,0 : REM <wh> to 0
1180
1190 DATA ***,*** A
```

```

161 start%=12+data%(&30)+256*
                        data%(&31)
162 DIM pal% start%-&38
163 PTR#f%=&38
164 FOR c%=1 TO start%-&38
165 pal%?c%=BGET#f%
166 NEXT
170 REM PTR#f%=12+data%(&30)+256*
                        data%(&31)
1111 PTR#f%=&38
1112 cnum%=0
1113 REPEAT
1114 VDU19,cnum%,BGET#f%,BGET#f%,
                        BGET#f%,BGET#f%
1115 VDU19,cnum%,BGET#f%,BGET#f%,
                        BGET#f%,BGET#f%
1116 cnum%+=1
1117 UNTIL PTR$f%=start% A
```

Converting View Files for First Word Plus

Clive Williams

Now that 1st Word Plus is available, many View users will be making the switch and their first concern will be try to recoup some of the time and effort they have invested in View over the years. View files can only be loaded from View into 1st Word Plus if they are pure ASCII text so it is necessary to spool the output from the SCREEN command or else use the WRITE command.

That is all very well but using a spooled file is less than ideal – re-formatting is a tedious chore due to all the hard spaces, etc., that are inserted into the text and the highlights are simply delimited by * and – characters which is frustrating when full WYSIWYG is available. The most satisfactory answer is to have a utility which takes View files that had been saved with the SAVE command and converts them to 1st Word Plus format as intelligently as possible.

Internal Formats

Before being able to write a conversion utility, some information is needed about how the two word-processors save text in their internal formats. View is easy: all the codes are given in the manual. This is not the case for 1st Word Plus, however, where a trial-and-error approach is needed.

A document which has been SAVED from View contains the following special codes:

- &09 – a TAB character
- &0B – a left margin TAB character (used when the left hand margin of the ruler is indented)
- &0D – end of line marker (carriage return)
- &1A – soft space
- &1C – highlight 1
- &1D – highlight 2
- &80 – followed by 2 characters = stored command
- &81 – followed by 2 dots = new ruler definition

The internal format of a 1st Word Plus file contains the following special codes:

- &0A – end of line marker (line feed)
- &0B – conditional page break
- &0C – unconditional page break
- &1B – starts a highlight code
- &1C – special formatting space

- &1D – starts a sequence of formatting spaces
- &1E – soft space (also space bar)
- &1F – starts a sequence of internal data
- &20 – hard space (also TAB space)
- &7F – TAB position in a ruler definition

A few notes on these codes will be useful to anyone writing conversion programs for other WP's.

The end of line marker is code &0A and it is treated as a hard end of line unless it is preceded with a character &1E.

The conditional page break code is followed by the number of lines concerned, incremented by 16 e.g. the sequence &0B &15 will cause a page break if there are fewer than 5 lines remaining on the page.

The highlight code, &1B, introduces a second byte which determines the highlight:

- &1B &80 – turns all highlights off
- &1B &81 – bold
- &1B &82 – light
- &1B &84 – italic
- &1B &88 – underline
- &1B &90 – superscript
- &1B &A0 – subscript

View uses codes &1C and &1D as toggles to switch on or off highlights 1 and 2 respectively. In 1st Word Plus, however, each highlight has a different bit value in the 8-bit highlight byte and highlights are turned on or off by setting or resetting the appropriate bit and the eighth bit is always set. It is important to note that multiple highlights are turned on or off in one byte e.g. to turn on both bold and underline in View, codes &1C and &1D are stored in consecutive bytes; in 1st Word Plus, the codes &1B &89 achieve the effect. To turn off underline, but leave bold, the codes &1B &81 are used. Whether a highlight is on or off is decided by EORing the highlight variable, htcod%, with the appropriate bit value – the first EOR will set the bit and turn the highlight on, a second one will reset it and turn the highlight off. The actual highlights which are selected by &1C and &1D are set by using the HT embedded command and different printer drivers make different use of this command.

The &1C code is used whenever a soft formatting space is required. It can be used in strings and the whole string is treated as a single entity.

The &1D code is used to introduce a string of soft formatting spaces at the start of a line e.g. following a centring or right-justification operation. Indents obtained by pressing <F9> are also stored as &1D followed by a string of &1C codes. Deletion of the leading &1D results in the string of &1C's also being deleted.

The code &1E is another sort of space which is obtained by pressing the space bar. This code is also used with &1C codes as soft formatting spaces in the middle of a line of text. Also, the sequence &1E &0A denotes a soft end of line marker.

&1F is used to introduce a string of internal data such as page margins, header and footer definitions, ruler definitions, etc. Some relevant examples are:

&1F &30 : introduces page layout information (see line 520)

&1F &31 : introduces header text (fields separated by &1F) (see line 530)

&1F &32 : introduces footer text (fields separated by &1F) (see line 540)

&1F &39 : introduces a ruler definition (see lines 430-450)

The ruler is enclosed within square brackets with &2E or &3A for character positions and &7F for TAB positions.

The ruler is terminated by 3 digits of which the first specifies the size of character: 0 – pica, 1 – elite, 2 – condensed, 3 – expanded

The second digit is a flag to indicate whether justification is on (1) or off(0).

The third digit is the line spacing information.

&1F &46 : introduces the footnote format data (see line 550)

(In addition, &18, &1F &45, &1F &4E and &1F &52 are all involved with footnotes but since this feature is not required to translate View files, they have not been fully investigated.)

The code for a hard space is &20. A string of &20's is inserted into the text whenever the TAB key is pressed and they can also be added using the pointer. Each hard space is treated as separate entity.

Difference between RF and Acorn versions

The only significant differences between the two

versions of the translation program are connected with the ways in the which the codes &1C and &1D, and the embedded command HT, are interpreted.

In the Acorn version of this program, the syntax of the HT command is: HT 2 130

Any other combination is not translated as a command and is saved in the file as a line of text and the extended highlight flag is switched off. Once the above code has been encountered, however, the extended highlights, as provided by the Acornsoft Printer Driver Generator, are available until the feature is switched off again.

Without extended highlights, codes &1C and &1D are used as normal to toggle underline and bold text respectively. The feature is unaffected by reaching the end of a line. With extended highlights, &1C turns on underline but it is automatically canceled at the end of the line. The other code, &1D, is used in combination with other codes to produce the various extended highlights. Those which are interpreted by this translation program are:

- : underline, automatically canceled at end of line

*- : subscript, automatically canceled at end of line

*-- : reset all highlights to off

** : superscript, automatically canceled at end of line

**_ : cancel subscript and superscript

_ : toggle italics on and off

*** : toggle bold on and off

The ones not supported, alternative font and extended character set, and invalid combinations, are saved as ordinary text. The two features not supported are too closely tied up with the 1st Word Plus printer driver (see files in the hex directory of the 1st Word Plus disc) to be sensibly included in this program.

In the RF version of this program, the HT command is used to select various highlights, other than underline and bold, by specifying a particular value which has been determined by the printer driver. Often, a large number of codes, giving full control of the printer, are available but in this version of the program, only 5 highlights are recognised:

HT 1 128 or HT 2 128 : select underline

HT 1 129 or HT 2 129 : select bold/emphasised

HT 1 130 or HT 2 130 : select italic

HT 1 135 or HT 2 135 : select superscript

HT 1 136 or HT 2 136 : select subscript

Any other combination is not translated as a command and is saved in the file as a line of text. If the command is valid, then it is saved in the variables ht1% or ht2% so that the next &1C or &1D refers to the correct highlight. The actual values associated with the highlights in View (i.e. 130, 135 and 136) are dependent on the printer driver and the DATA statements at the end of the program, where this information is stored, will need to be altered for different printer drivers. In these DATA statements, the first digit is a 1st Word Plus code and must not be changed. The second number, however, is the code which is used to obtain each of the five supported highlights from View and it is quite likely that users will need to change some of these numbers to the ones used by their printer driver software (the underline and bold codes, 128 and 129, are the default condition and probably will not need to be changed).

The Program

The program begins by initialising some variables in 'PROCinit'. Users may wish to alter some of these variables – in particular:

size\$="1" gives elite (12 cpi) text

justify\$="1" selects justification on

linespace\$="1" selects single line spacing

These values are declared on line 100 and they are used whenever a ruler is declared. Clearly, they may not suit everyone's needs. On the same line, a logical variable called 'pound' is set to TRUE. In this state, whenever the program encounters the code for a pound sign (declared in the last DATA statement of the program), it changes the code to &A3 which is the 1st Word Plus code for a pound. If 'pound' is declared as FALSE, then no such changes occur.

After declaring the main variables and setting up various arrays, the program starts to create the 1st Word Plus destination file. When 1st Word Plus is put into word-processing mode by selecting 'WP mode' from the 'Edit' option, a header, in internal format, is placed at the start of the file. This header contains default settings for page layout, headers, footers, and footnotes and a default ruler is set up. This operation is performed by 'PROCheader'.

The rest of the program is reasonably easy to follow. Each line of the View file in turn is read into the variable, line\$. If this begins with a stored (embedded) command it is decoded in 'PROCembedded', whereas if it is a new ruler definition, it is decoded in 'PROCnew_ruler', otherwise it is sent straight to 'PROCwriteline' to be written to the new file.

Each byte of the data to be output is checked to see if it is a special character used by View's internal format – if so then appropriate action is taken. The program makes a fair attempt to interpret the View file but when it meets a situation it cannot cope with, a flag is set and a helpful message is displayed after the conversion process is complete.

(Only one of the programs is listed because of space problems. Both are on the monthly program disc.)

```

10 REM >$1ST_WP.V1W+_Acorn V 1.1
20
30 REM Version for Acornsoft
   Printer Driver Generator
40 REM By Clive Williams
50 REM 19.5.1988
60
70 REM Not implemented:Alternative
80 REM fonts & extended character
90                                     set
100 size$="1":justify$="1":
    linespace$="1":pound=TRUE
110 CLOSE#0
120 MODE 12:OFF:VDU 19,0,4,0,0,0,19,
    1,6,0,0,0
130 PRINTTAB(20,2);"View to 1st
    Word+ File Conversion"
140 PROCinit
150
160 INPUTTAB(0,5);"Name of VIEW
    file";view$
170 INPUTTAB(0,6);"Name of 1st Word+
    file";firstword$
180 view%=OPENIN(view$)
190
200 IF view%<>0 THEN
210 PRINTTAB(0,7);"Converting line ";
220 firstword%=OPENOUT(firstword$)
230 PROCheader
240 WHILE NOT EOF#view%
250 line$=FNreadline
260 IF line$<>"" THEN
270 CASE ASC(LEFT$(line$,1)) OF

```



```

280     WHEN &80:PROCembedded
290     WHEN &81:PROCnew_ruler
300     ENDCASE
310     ENDIF
320     PROCwriteline(line$)
330     ENDWHILE
340     CLOSE#0
350     PROCmessages
360     ENDIF
370     END
380
390     DEFPROCinit
400     line%=0:lf$=CHR$(&0A):htcode%
        =&80:lm%=0
410     printfile=FALSE:addruler=FALSE:
        pagelayout=FALSE:hilite=FALSE
420     LMcommand=FALSE:header=FALSE
        :extended=FALSE
430     ruler%=74:r$="....."+CHR$(0)
440     ruler$=STRING$(9,r$)+".."
450     r$=CHR$(&1F)+"9["+ruler$+"]"+
        size$+justify$+linespace$+lf$
460     READ pound%
470     ENDPROC
480
490     DEFPROCheader
500     LOCAL header$
510     header=TRUE
520     header$=CHR$(&1F)+"0"+
        "6601030305800"+lf$:
        PROCwriteline(header$)
530     header$=CHR$(&1F)+"1"+CHR$(&1F)
        +CHR$(&1F)+lf$:PROCwriteline
        (header$)
540     header$=CHR$(&1F)+"2"+CHR$(&1F)
        +"##"+CHR$(&1F)+lf$:PROCwriteline
        (header$)
550     header$=CHR$(&1F)+"F0110030"+
        lf$:PROCwriteline(header$)
560     PROCwrite_ruler
570     ENDPROC
580
590     DEFFNreadline
600     LOCAL a%,a$
610     line%+=1
620     PRINTTAB(16,7);STRING$(5," "):
        PRINTTAB(16,7);line%
630     a%=BGET#view%
640     WHILE a%<>&D
650     a$=a$+CHR$(a%)
660     a%=BGET#view%
670     ENDWHILE
680     =a$

700     DEFPROCwriteline(a$)
710     LOCAL i%,j%,k%,char%,pos%,
        string$,temp%
720     IF a%<>" THEN
730     pos%=0
740     IF lm%<>0 AND (LEFT$(a$,1)=
        CHR$(&0B) OR LMcommand) THEN
750     header=FALSE
760     pos%=lm%:i%=0:BPUT#firstword%
        ,&1D
770     WHILE i%<lm%
780     BPUT#firstword%,&1C:i%+=1
790     ENDWHILE
800     ENDIF
810     FOR i%=1 TO LEN(a$)
820     char%=ASC(MID$(a$,i%,1))
830     CASE char% OF
840     WHEN &01 :BPUT#firstword%,
        &0B:i%+=2:char%=ASC(MID$(
        a$,i%-1,1))
850     WHEN &09 :
860     IF INSTR(MID$(ruler$,pos%+1)
        ,CHR$(0))<>0 THEN
870     WHILE MID$(ruler$,pos%+1,1)
        <>CHR$(0)
880     BPUT#firstword%,&20
890     pos%+=1
900     ENDWHILE
910     ENDIF
920     char%=&20:pos%+=1
930     WHEN &0A :IF extended THEN
        PROCturn_off
940     WHEN &0B :char%=&00:pos%+=1
950     WHEN &1A :char%=&1E:pos%+=1
960     WHEN &1C :
970     BPUT#firstword%,&1B
980     htcode%=htcode% EOR 8
990     IF NOT extended AND MID$(a$,
        i%+1,1)=CHR$(&1D) THEN
1000     i%+=1:htcode%=htcode% EOR 1
1010     ENDIF
1020     char%=htcode%
1030     WHEN &1D :
1040     IF extended THEN
1050     string$="*"
1060     j%=ASC(MID$(a$,i%+1,1))
1070     WHILE j%=&1C OR j%=&1D
1080     i%+=1
1090     IF j%=&1C THEN
1100     string$=string$+"-"
1110     ELSE
1120     string$=string$+"*"
1130     ENDIF

```

View to First Word Plus

```

1140     j%=ASC(MID$(a$,i%+1,1))
1150 ENDWHILE
1160 CASE string$ OF
1170     WHEN "***":htcode%=htcode%
                                OR &10
1180     BPUT#firstword%,&1B:char%=
                                htcode%
1190     WHEN "****":htcode%=htcode%
                                EOR 1
1200     BPUT#firstword%,&1B:char%=
                                htcode%
1210     WHEN "***":htcode%=htcode%
                                OR &20
1220     BPUT#firstword%,&1B:char%=
                                htcode%
1230     WHEN "***-":htcode%=htcode%
                                AND &CF
1240     BPUT#firstword%,&1B:char%=
                                htcode%
1250     WHEN "***---":htcode%=&80
1260     BPUT#firstword%,&1B:char%=
                                htcode%
1270     WHEN "***-":htcode%=htcode%
                                EOR 4
1280     BPUT#firstword%,&1B:char%=
                                htcode%
1290     OTHERWISE:FOR j%=1 TO LEN
                                (string$)
1300     BPUT#firstword%,ASC(MID$
                                (a$,j%,1)):pos%+=1
1310     NEXT
1320 ENDCASE
1330 ELSE
1340     BPUT#firstword%,&1B
1350     htcode%=htcode% EOR 1
1360     IF MID$(a$,i%+1,1)=CHR$(&1C)
                                THEN
1370     i%+=1:htcode%=htcode% EOR 8
1380     ENDIF
1390     char%=htcode%
1400     ENDIF
1410     WHEN &20:char%=&1E:pos%+=1
1420     WHEN pound%:IF (pound AND NOT
                                header) char%=&A3:pos%+=1
1430     OTHERWISE:pos%+=1
1440 ENDCASE
1450     IF char%<>0 THEN BPUT#
                                firstword%,char%
1460 NEXT
1470 ENDIF
1480 IF a$="" THEN
1490     BPUT#firstword%,&0A
1500 ELSE
1510     IF RIGHT$(a$,1)<>lf$ THEN
1520     IF extended THEN PROcturn_off
1530     BPUT#firstword%,&1E:BPUT#
                                firstword%,&0A
1540     ENDIF
1550     ENDIF
1560     ENDPROC
1570
1580     DEFPROcturn_off
1590     htcode%=htcode% AND &87
1600     BPUT#firstword%,&1B
1610     BPUT#firstword%,htcode%
1620     ENDPROC
1630
1640     DEFPROCembedded
1650     LOCAL i%,code$,validcode,ht%
1660     line%=MID$(line$,2)
1670     code%=LEFT$(line$,2)
1680     line%=MID$(line$,3)
1690     CASE code$ OF
1700     WHEN "PE":
1710     i%=VAL(line$)
1720     IF i%=0 THEN
1730     line%=CHR$(&0C)
1740     ELSE
1750     line%=CHR$(&01)+CHR$(i%+&10)
                                +lf$
1760     ENDIF
1770     WHEN "LM":
1780     lm%=VAL(line$):line$=""
1790     IF lm% THEN LMcommand=TRUE
                                ELSE LMcommand=FALSE
1800     WHEN "CE":
1810     IF ruler%>LEN(line$) THEN
1820     BPUT#firstword%,&1D
1830     FOR i%=2 TO (ruler%-LEN(line$
                                )) DIV 2
1840     BPUT#firstword%,&1C
1850     NEXT
1860     ENDIF
1870     line%=line$+lf$
1880     WHEN "LJ":
1890     WHILE LEFT$(line$,1)<CHR$(&21)
1900     line%=MID$(line$,2)
1910     ENDWHILE
1920     line%=line$+lf$
1930     WHEN "RJ":
1940     IF ruler%>LEN(line$) THEN
1950     BPUT#firstword%,&1D
1960     FOR i%=2 TO ruler%-LEN(line$)
1970     BPUT#firstword%,&1C
1980     NEXT
1990     ENDIF

```



```

2000 line$=line$+lf$
2010 WHEN "HT" :
2020 IF LEFT$(line$,1)=" " THEN
2030 REPEAT:line$=MID$(line$,2)
      :UNTIL LEFT$(line$,1)<>" "
2040 ENDIF
2050 IF LEFT$(line$,5)="2 130" THEN
2060 extended=TRUE:line$=MID$
      (line$,6)
2070 ELSE
2080 extended=FALSE:line$=code$+
      " "+line$+lf$:hilite=TRUE
2090 ENDIF
2100 WHEN "TS","SR":printfile=TRUE
      :line$=code$+" "+line$+lf$
2110 WHEN "LS":addruler=TRUE
      :line$=code$+" "+line$+lf$
2120 WHEN "DH","DF","HE","FO","PL"
      ,"TM","HM","FM","BM"
      :pagelayout=TRUE:
      line$=code$+" "+line$+lf$
2130 OTHERWISE:line$=code$+" "+
      line$+lf$
2140 ENDCASE
2150 ENDPROC
2160
2170 DEFPROCnew_ruler
2180 LOCAL i%,char%
2190 line$=MID$(line$,4):ruler$=""
2200 i%=0
2210 REPEAT:i%+=1:UNTIL MID$(line$
      ,i%,1)=">" OR i%>LEN(line$)
2220 IF i%>LEN(line$) THEN lm%=0
      ELSE lm%=i%-1
2230 i%=LEN(line$)+1
2240 REPEAT:i%-=1:UNTIL MID$(line$
      ,i%,1)="" OR i%=0
2250 IF i%>0 AND i%>lm% THEN line$
      =LEFT$(line$,i%)
2260 FOR i%=1 TO LEN(line$)
2270 CASE MID$(line$,i%,1) OF
2280 WHEN "*, ">":ruler$=ruler$
      +CHR$(0)
2290 WHEN ".","<":ruler$=ruler$+"."
2300 ENDCASE
2310 NEXT
2320 ruler%=LEN(ruler$)
2330 r$=CHR$(&1F)+"9["+ruler$+"]"+
      size$+justify$+linespace$+lf$
2340 PROCwrite_ruler
2350 ENDPROC
2360
2370 DEFPROCwrite_ruler
2380 FOR i%=1 TO LEN(r$)
2390 char%=ASC(MID$(r$,i%,1))
2400 IF char%=0 THEN char%=&7F
2410 BPUT#firstword%,char%
2420 NEXT
2430 line$=""
2440 ENDPROC
2450
2460 DEFPROCmessages
2470 PRINT"Conversion now complete";
2480 IF printfile OR addruler OR
      pagelayout OR hilite THEN
2490 PRINT" but you may find the
      following notes useful:"
2500 ELSE PRINT"."
2510 ENDIF
2520 IF printfile THEN
2530 PRINT"You have used the TS or
      SR embedded commands in your
      document. In 1st word+"
2540 PRINT"these are set at the
      time of printing ie. when you
      select 'Print File' from"
2550 PRINT"the 'File' option. If
      the HT command has been used to
      select NLQ printing, then"
2560 PRINT"this also is selected
      within the printing option."
2570 ENDIF
2580 IF addruler THEN
2590 PRINT"You have used the LS
      embedded command to alter line
      spacing. In 1st word+, this"
2600 PRINT"is done by defining a
      new ruler. Therefore, you will
      need to select 'Add Ruler'"
2610 PRINT"from the 'Layout' option
      in the main menu. Similarly,
      a new ruler must be set"
2620 PRINT"up if you have used a HT
      command to change the character
      pitch:Pica, Elite,"
2630 PRINT"Enlarged and Condensed
      are all supported in this way."
2640 ENDIF
2650 IF pagelayout THEN
2660 PRINT"You have used embedded
      commands to alter the header,
      footer, page length or"
2670 PRINT"margins (ie. DH,DF,HE,FO
      ,PL,FM,TM,BM or HM). 1st word+
      allows these to be set"

```

```

2680 PRINT"up within the 'Page
      Layout' option which is within
      the 'Layout' option in the"
2690 PRINT"main menu."
2700 ENDIF
2710 IF hilite THEN
2720 PRINT"You have used the HT
      embedded command to select a
      printing function which is"
2730 PRINT"unknown to this program.
      If it is for underline, bold,
      italic, emphasised,"
2740 PRINT"superscript or subscript
      , then amend the data lines at
      the end of this program"
2750 PRINT"to suit your VIEW
      printer driver."

```

```

2760 PRINT"If it is to select
      different sized characters, then
      you will need to define a"
2770 PRINT"new ruler by selecting
      'Add Ruler' from the 'Layout'
      option."
2780 PRINT"If it is to select NLQ
      (Near Letter Quality), then
      this is an option which can"
2790 PRINT"selected within the
      'Print File' option of
      1st Word+."
2800 ENDIF
2810 ENDPROC
2820
2830 REM Pound sign
2840 DATA 35 A

```

Using the ARM BASIC Editor

Tim Saxton

Built into the ROM software on the Archimedes is a very powerful screen-based editor for BASIC programs. The operation of this is quite different from the line editor on the Model B. There are similarities with the Editor in the Master, but the Archimedes ARMBE far surpasses it in all respects. Because it is so different from the old line editor, which of course has been reproduced for compatibility in the new machine, some users may not have started to use the new Editor. I hope that this article will encourage you to try it and, having once done so, I am sure you will never again use the line editor, which is totally eclipsed by ARMBE.

Getting Started

The editor can only be entered from BASIC and an error is generated if you try to use *ARMBE from the Arthur prompt. From BASIC, either EDIT or *ARMBE will transfer to the editor. If there is a BASIC program in memory, the editor will display the first part of it and, if not, it will show a clear (blue) screen and offer the first line number of 10.

If you know the line number you want to edit then append this number to the EDIT command – e.g. EDIT 950 will enter the editor with line 950 displayed at the top of the screen. If the number cannot be found, the next line after will be put at the top of the screen. Alternatively, typing something like EDIT FRED will enter the editor with the line

containing the first occurrence of FRED at the top of the screen. (See Hints & Tips, page 6.)

Entering the editor does not clear all the variables, so it is possible to run a program, inspect it with the editor, exit the editor and still use LVAR to view all the variables. Of course, if you make the slightest change in the edit mode, everything is reset.

Status

You will notice a white single row window at the bottom of the screen. This is the status line and shows various pieces of information about the program being edited.

The number on the left is the size of the program being edited, in bytes. This is updated each time you move from one line to another.

The word next to the size number will either be 'Modified' or 'Original'. This tells you if you have made any change to the program since you last entered the editor or saved the program. If you save the program from within the editor using <f3> then this indicator is reset to 'Original'. The next entry on the status line is the current file name being used for the program. If the copy option is active, this is shown and various prompts for searches, etc will also appear on the right hand side of this window

Moving Around the Program

The position currently being edited is shown by the cursor. This is a rectangle in insert character mode

or an underline in overwrite mode. Whether this is flashing or not can be set as an option—see later. The mouse is not used at all by the editor, all cursor movements being controlled by the keys. Pressing <ctrl-f5> then <space> shows the cursor movement options available. Additionally some functions are duplicated on the number keypad on the right of the keyboard when the 'num lock' light is out. The list below shows these extra key functions:-

Move down a page	3
Move up a page	9
Move down a line	2
Move up a line	8
Move right one character	6
Move left one character	4
Present line to top of screen	7

Alternatively, the line number to move to can be specified by pressing <ctrl-f12> when a window is produced asking for the required line number.

Returning to a particular part of the program is achieved using <f6> to 'mark' the line (a "." appears to the right of the line number) and then when you need to return to it, press <shift-f6>. There may be up to 4 marked lines and repeated presses of <shift-f6> cycles round them. A mark is removed by putting the cursor in the required line and pressing <f6> again. The second page of Info, <ctrl-f6>, shows the line numbers with marks set.

It is worth noting that you cannot get at the line number in any way at all. The editor is in charge of all line numbers and the cursor cannot be moved to them. The only influence you can have is to renumber the program, if you wish, by using <f8>, although Archimedes BASIC makes line numbers all but superfluous.

If, during a renumber, a line reference cannot be found (maybe you have deleted the line) then the line number reference is replaced with '@@@@' and an error window appears. Use <f4> or <f7> (search functions, see later) to find the lines containing '@@@@' to correct them.

Editing Within a Line

Having moved to the part of the program you want, the next step is to edit. Inserting characters at the cursor is the default and the keyboard is used in the normal way. Alternately, overtyping is available by pressing the <insert> key—note the change in cursor

style. (This key may be duplicated by the keypad '0'.) Pressing <insert> again resets to insert mode.

Inserting a New Line

Pressing <return> at any point will leave the current line and insert a new blank line after it, with the cursor at the first character. The line numbers used are all sorted out by the editor and if there isn't room, numerically speaking, for your new line, the whole program is automatically renumbered. BASICally, forget line numbers!

Deletion Within a Line

Pressing <delete> (or keypad '.') eliminates the character to the left of the cursor and moves everything left one character. Pressing <shift-delete> eliminates the character under the cursor and moves all the characters to the right of the cursor by one character towards the left. <f11> deletes from the cursor to the end of the line and <shift-f11> deletes from the cursor to the start of the line. <ctrl-f11> removes all characters from the line, but retains the line number. To completely eliminate the line, number and all, mark it by positioning the cursor somewhere in it and press <ctrl-D> (for delete) and then <f12>.

Restoring an Edited Line

A useful facility is the 'undo' function, <shift-f10>, which will put a line being edited back to the state it was when the cursor was moved to it. However, this will only remember the previous state of the line **until the cursor is moved away from the line**. It then of course records the initial state of the line it has moved to. This means that you can only recover a particular line while the cursor is still on the line. Check before you move off!

Copying a line

There are three ways of doing this. (1) If an exact copy immediately following the current line is required then 'repeat', <shift-f8> will fill the bill. (2) If the whole line is to be copied somewhere else, mark the line using <ctrl-C> (for copy) move the cursor to the line before or after the required destination and mark this line with <ctrl-A> (after) or <ctrl-B> (before) as appropriate. Now pressing <f12> (execute) will carry out the operation. (3) (Use this method with care—it contains the only bug that I have found so far in the editor.) If you need to copy a part of a line to somewhere else on the visible

screen then the 'copy' key (or keypad '1') may be used. Firstly position the cursor where you want the copy to go. Then press <copy>. Now move the cursor to the first character you want to copy. Press <copy> for each character you want (hold it down for auto-repeat). When you have finished, press <escape> or <return> if you want a new line next. The bug occurs when you are copying into a line ABOVE the one you are copying from. If you overflow the end of a line as you add characters then the whole screen below this moves down a line, but the copy cursor does not, so you start to copy the wrong line! It isn't difficult to avoid the situation, but it is most confusing when it happens.

When working within a line, you may cause blank lines to appear as you delete characters to reduce the overall length below 80 or so. Don't worry, when you move the cursor from the line the editor sorts it all out and reformats the screen.

Line operation commands

We have seen already some of the powerful complete line operation commands (<ctrl-A>, -B>, -C>, -D>). Let's look at this whole set. Firstly a few general rules and features.

- Only one complete operation may be carried out at a time.
- The source used for the operation may be one line (a single character marking that line) or a group of contiguous lines (a double character marking the first line in the group and the same double character marking the last line).
- A marked line is indicated by one or more capital letters in place of the line number. (The line number isn't lost, just hidden.)
- Any mark can be removed by putting the cursor in the line and pressing <ctrl-R> (remove).
- All the character marks can be removed by pressing <ctrl-R> when the cursor is in an unmarked line. (Note that "." marks are not affected by this.)
- Presses on <shift-f12> will cycle round the lines with line commands set, putting the line at the top of the screen.
- The second Info page (<ctrl-f6> followed by <page-down>) will show the line commands set and the corresponding line numbers.

- When a program is saved, it is not necessary to remove any of the marks, they are ignored by the save operation.

- If you try and do anything silly or illegal, an error window will appear giving more or less helpful advice. Press <escape> to recover from this condition.

Source line or block identifiers:-

M or MM	Move line or block
C or CC	Copy line or block
D or DD	Delete line or block
JJ	Justify block
L or LL	Label line or block

If you use the double character form, they must appear on two lines, the first being the start of the block, the second being the end.

Destination identifiers:

T	Top of program (i.e. before existing first line)
E	End of program (i.e. after existing last line)
A	After this line
B	Before this line

A few examples may help to show how easy it really is. Let us suppose we have a program with lines from 10 to 1000 in steps of 10. To move line 90 after line 250. Position the cursor anywhere in line 90 and press <ctrl-M>. Position the cursor anywhere in line 250 and press <ctrl-A>. (The order of these two operations is unimportant.) Then press <f12>. Line 90 vanishes and reappears as line 251. (The same effect could be obtained by replacing the second step with a <ctrl-B> in line 260.)

To make a copy of lines 400 to 500 inclusive after line 500. Go to line 400, press <ctrl-C><ctrl-C>. Go to line 500, press <ctrl-C><ctrl-C><ctrl-A>. Press <f12>. Lines 400 to 500 are untouched, but copies of them are put in after 500. This would incidentally cause an automatic renumber to take place, so the first line copied will be 510, etc and the old line 500 will now be 620.

You can see that line numbers now really are no good for remembering where things are. OK for the computer, which just adjusts its references, but that little routine that started at say, 500 has now become 620! - REM statements are a must!

Do not try to move or copy a block to within itself!!

Labels (L and LL) are used to limit saving, searching and replacing to certain parts of the program..

File Operations

Saving a file (<f3>) presents a window, with the default program name taken from the first line of the program if it has a "REM> filename" on it. <Return> overwrites any existing file with the program in memory. If you want to use a different name, <f11> will clear the line before you enter the new name. If either L or two LL's are set then you are "bonged" and a warning that you are only saving a limited part of the program is given.

Loading a file (<f2>) presents a similar window, but always needs a name and overwrites the existing program in memory (if any) with the one specified. If you have a modified unsaved program in memory, a message and "bong" remind you. <Escape> will leave the load operation. In fact, <escape> will abort any operation.

Appending a file, <shift-f2>, adds a new program onto the end of the one already in memory. The line numbers are all looked after by the editor, the appended program line numbers being adjusted to follow on at the end of the already resident program.

A warning about loading and saving files: The window for these two operations appears very similar in the heat of programming and the selection keys are next to each other. If you want to load a program (<f2>) but hit <f3> in error, type in your program name and hit return, the program you want will be overwritten by whatever you were editing before! If it was the start of a session, then your precious program will still have an entry in the directory, but loading it will reveal nothing... Of course it shouldn't happen, but a couple of programs 'disappearing' makes me realise how easy it is to do.

Window Facility

It is often useful to be able to see one part of a program while editing another part - <ctrl-f4> allow this to be done. The screen is split, with the cursor in one half. The active half with the cursor will now act exactly like the full window, scrolling and editing as before. Any changes carried out globally will be reflected in the other window. To swap to the other window press <ctrl-f2> and the cursor moves to the other window, with full control.

Each time <ctrl-f2> is pressed, the cursor swaps windows, returning to its previous position in each window. <ctrl-f1> pressed a second time joins the screen into one window. It is not necessary to join the windows for other editor operations, e.g. save.

Search and Replace

<f5> or <shift-f5> allow searches of the program for strings and keywords. Keywords are entered as they appear (e.g. PRINT). The editor remembers the last keyword used and if you want a new search word, use <f11> to clear the search field prior to a new entry. The selective replace prompts for each occurrence in the bottom window, while global replace changes every occurrence once <return> is pressed after the new string is entered. The line labels 'LL' allow a limit to be placed on the part of the program modified by the search facilities.

<f7> will search for a string and list to screen every line it appears in. If there is more than one screenful then <shift-f7> will step down the screens. It is not possible to step backwards through the screens with <ctrl-f7>. When in the search mode, no changes may be made to lines and the cursor cycles round the visible screen only. <Escape> returns to the normal edit mode. The most useful function of this mode is to enable you to list the lines containing the search word, find the ones you want to alter, mark up to 4 of them with a "." (<f6>), then cycle through them for editing with <shift-f6>. If you have a lot of lines to change, then <f4> is a better choice.

<f4> will search for the occurrence of a string and put the cursor on the start character for you to work on. <shift-f7> and <ctrl-f7> move down and up the occurrences. It is possible to do all sorts of editing at an occurrence, almost without limit and still step up or down from that occurrence to any other.

Miscellaneous Bits and Pieces

* commands can simply be entered using <f1>. The star prompt being re-offered until a <return> on its own is entered. Care should be exercised with these very powerful commands, of course, or a 'crash' or other undesirable effect will occur.

Splitting & joining lines. <shift-f1> will split a line at the cursor. The character under the cursor will be the first character on the new line. <ctrl-f1> joins the line with the cursor to the next line after. The ":" separator is automatically added.

Options, <shift-f3> lets you set various features in the editor, such as cursor flashing, colours, tab size, etc. The cursor controls select the feature you want to alter, then <space> will cycle round the options. When the right one is shown, <escape> returns to edit. The value of all these options is stored in non-volatile RAM, so next time you invoke the editor the settings will be the same.

The few f-key commands left are really self explanatory and complete a really impressive editor which must have doubled my productivity when writing BASIC programs. If Acorn carry on improving the facilities and sophistication of BASIC, I never will get round to using any of the compiled languages! **A**

A Spell-Checker for all WP's?

Adrian Look

Computer Concepts' Spell-Master can be made to work on files from all sorts of word-processors. When combined with Archive's Spell-Master Utilities Disc, it can be used to check Graphic-Writer, Arc-Writer, Wordwise Plus and, of course, Inter-Word texts.

Graphic-Writer and Arc-Writer

Spell-Master's Checkfile command works with both Graphic-Writer and Arc-Writer files but you have to change the file types:

*SETTYPE filename FFF <return>

*CHECKFILE filename <return>

*SETTYPE filename 400 <return>

(or *SETTYPE filename AF0 with Arc-Writer)

This seems to keep all the formatting intact but all it does is to insert a # at the beginning of each word it does not recognise! Computer Concepts have now changed it so that the command allows you the option of using a different character instead of # – this is used as a wild card on some word-processors and is therefore rather difficult to find!

However, if you use the much more versatile spelling checker implemented on the Archive Spell-Master Utilities Disc, you can have user dictionaries, add words, ignore words, browse the dictionary, edit the word etc as you would expect to do if the spelling-checker were integral with the word-processor and you don't have to worry about the filetype – that is taken care of by the program.

Spell-Master on Wordwise too!

*Checkfile also works with Wordwise Plus files, but the settype doesn't need changing since it is &FFF already. Again, the Archive spelling checker can be used for greater versatility.

Next month we will look at the SYS commands provided by Spell-Master which provide an extremely powerful add-on for programming, in any language, that involves the use of words. The Check and Anagram commands, for example, can be made to return words that can be used within a program. This makes crossword (and other word-game) generating programs a real possibility. It could also be used for validating user input – if the user inputs, say, the word "reciept", the program can SYS"Check" it and, finding it unknown, SYS "Typo" it and offer the resulting word to the user, "Did you mean receipt?". Then with user dictionaries as well, you can add specialist responses.

The list of possible applications is endless. If there is something you would particularly like to see implemented in this area, let me know through the editor and we'll see what we can do.

Archive Spell-Master Utilities Disc

If you buy the Computer Concepts Spell-Master ROM through Archive, you get a free utilities disc:

- A dictionary for use with word-games. (I can't call it a Scrabble Dictionary.)
- Spell-check utility including 1 & 2 letter words which should work on any WP files (hopefully, but let us know!)
- Anagram program – generates as many words as possible from combinations of given letters
- Program to direct output of *-commands to files so that you can use the lists of words generated for other applications

(Also included are two Wordwise Plus programs, originally written for BBC Spell-Master, in case they are of interest to anyone.)

*If you have bought Spell-Master elsewhere (shame on you!) you can buy the Utilities Disc for £5. **A***

PipeDream – The Ultimate in Integration?

Gerald Fitton

The claim for PipeDream is that "No other integrated software gives you the power to blend text and numbers so completely". The more I use PipeDream the more I have come to believe this claim of Colton Software. There is no need to create text in one window and carry out calculations in another. No need for special "listing documents" in order to read the fields of a database. All the features of the software, wordprocessor, spreadsheet and database can be on the screen at the same time, in the same document, without the need to cut and paste, exactly as they will be printed. PipeDream files do not discriminate between WP, spreadsheet or database. At £113.85 PipeDream is a bargain.

How Easy is it to Get Started?

You must make a copy of the disc before you do anything else. Then follow the simple instructions given in the first 18 pages of the manual, which explain how to install the software on your machine. Type PD /12 to load PipeDream then press the <Alt> key to gain access to the easy-to-use pull down menus and you are away. Beginners will have few, if any, problems.

The Word Processor

When you start up PipeDream you will see theA.....B etc of a spreadsheet where you might expect to see a ruler. You will find row numbers appearing as you type in text. However, unlike a spreadsheet, your text is not limited to the width of the spreadsheet column – you can use the whole width of the page. Nevertheless, so far as the sheet is concerned you are in column A, right up to the place where you set your right margin, default 71.

Using PipeDream simply as a wordprocessor, you will find you have the usual standard features expected of a modern powerful WP package. They are available from the function keys, the <Alt> key and from pull down menus. The novice will probably find the pull down menus most useful whereas the more experienced will quickly learn to use the <Alt> key sequences. These include block copy, search and replace, case changes etc. together with a paragraph reformat function. I found the best

way of varying the left margin within a document was to set column A to a suitable width and then <Tab> to column B for the inset material. Column B appears to the user as a semi-transparent overlay of column A which becomes opaque only when text is typed into B over the top of text placed in A. Text in multiple columns is easily achieved with facilities such as individual column reformat, individual right margin settings and block copy between columns. You can add a document, or even part of a document to one already in memory. By fixing rows, you can hold part of a document on the screen while the rest is scrolled by. This is useful if you need to copy parts of one document into another. I found it easier than using the multiple windows offered by some other WP packages.

Printing

Printer Drivers are supplied for the Juki 6100 daisywheel and the Epson FX80. About a dozen pages of the 325 page manual are devoted to the subject of creating printer drivers. If, for example, your printer supports microspacing, foreign characters, bold, underline, italics, superscripts, etc. then you might need to study this section to get the best out of the package. However, you can probably use one of the printer drivers supplied to start with. The output is truly WYSIWYG with a good screen representation of bold, underline, etc.

The Spreadsheet

As a spreadsheet, PipeDream uses theA.....B etc. columns, which can be individually set to any width. It contains all the usual operations such as insert or delete row and column, number format settings, manual or auto recalculate as well as a comprehensive replicate. Snapshot replaces a slot expression with its current value; text can be converted into expressions, and vice-versa with a single keystroke. The range of expressions include the usual arithmetic ones, trigonometrical, business and statistical functions. In addition, there are facilities for using logical operators, conditional expressions, lookup tables and for reading from and writing to files. Rows or columns from one spreadsheet can be merged with

another held in memory. Rows or columns can be "fixed" whilst the rest of the sheet is scrolled past the fixed slots; this makes the copy option simplicity itself. Dates can be entered in UK or American format. Although the spreadsheet area is quoted as 536,870,912 (yes, half a billion!) each way, in reality it will be limited by the memory of the machine. Leading or trailing characters, such as £ or % signs can be added to any slot without affecting calculability. Files from Lotus 123 can be exchanged with PipeDream. Perhaps the feature that users of other spreadsheets will notice more than anything else is the speed, but what else would you expect? — It is an Archimedes, after all!

The Database

The spreadsheet columns double as fields for the database so you can have up to half a billion fields per record and, maybe, half a billion records per file! Of course it is limited by the amount of memory in your machine. The sort facility is elegant if simple. The sort can be numeric, alphabetic or chronological. In conjunction with the option to save selected rows as a file, sort provides a powerful file splitting feature. Search and replace "borrowed" from the WP or Replicate from the spreadsheet can be used on any database field. Columns can be interchanged with a block move and the result printed as it appears on the screen. Free text can be added in separate rows to annotate the fields or even as comments between sets of records. A database file, when saved in a suitable format, will provide data fields which can be merged with a form letter for mail shots or invoicing.

Integration

What is outstanding about PipeDream is the degree to which the WP, spreadsheet and database are integrated. It is true that you can not see the join. The reason is that there is none to find! This is not just another integrated package consisting of three separate programs. All the features of the software are available whether you are dealing with text fields, numeric fields, mathematical expressions, free text or, quite often, all of these within the same document. For example, you can introduce any of the "spreadsheet" expressions into the middle of a paragraph of free text. Of course, with this degree of integration, you would expect and indeed get a printout of exactly what you see on the screen.

For the Expert

The expert will soon become familiar with the <Alt> key commands, but PipeDream also offers customisation. The experienced user can redefine the keyboard, reconfigure the printer driver and, with macro files saved to and read from disc, execute a sequence of commands as if they were entered from the keyboard.

File Compatibility

PipeDream is available for the Z88 portable and for MS-DOS machines (PCs). The files are compatible with those of View Professional. I borrowed a Z88 and transferred a few Z88 PipeDream files. File transfer worked perfectly and the Z88 owner was most impressed by the speed of the Archimedes spreadsheet. I was so impressed by the way PipeDream works on the Z88 I might even buy one! We hear a lot these days about making one machine accept software designed to run on another machine. The PC Emulator for the Archimedes is only one of many such emulations. However, what the user is interested in is applications, not operating systems. Someone who has learned how to use PipeDream on a PC will be able to use the Archimedes or Z88 version immediately and transfer files between them with no messy conversions.

Instead of printing out a PipeDream file from the Archimedes, output can be directed to a file. This review was created in PipeDream and then (because I know Paul Beverley likes it) directed to a file which I loaded into Wordwise. The problems were trivial (e.g. the sign for pounds "£"). Lotus files can be interchanged, Lotus macros are preserved by PipeDream and can be sent back to Lotus unchanged but they are not usable within PipeDream. I spooled out a large data file from ViewSheet, the monthly employment statistics for about 30 years, loaded it into PipeDream and set up the usual time series spreadsheet with 12 month moving averages and seasonal variation. The whole exercise took about ten minutes and worked beautifully. I then added some heading text and selected the portion I wanted to print. I had no problems at all!

Limitations

What is missing? Well, if I have to find something, it is the inability to insert graphics within the text in

the way that say Graphics Writer will. A 'phone call to Colton Software confirmed what I suspected: they are working on this upgrade but it might be a little while before it comes on the market.

Summary

Colton Software have produced a software package which fulfils all the requirements of a fast spreadsheet, easy to use database and powerful wordprocessor. We need a new word to describe this degree of integration: there is only one type of file, a document containing text, data, numbers and expressions, the commands usually peculiar to one type of field can be used on any other. The beginner will find it easy to get started and the experienced will be able to use the advanced features to customise or get even more speed. The document files are transportable to the Z88 and to PCs. On these other machines the same applications software is available. **A**

Contact Box

- K Faulkner has the last 18 years of football results on the Archimedes in a ram database. Anyone interested in statistical analysis of such data or has an interest in the subject, please contact him at 8 Ardmore Heights, The Brae, Ballygowan, Co Down, BT23 5TU.
- The Fractal Explorers' Club is a national, non-profit making computer graphics club run by John Catchpole, 5 Carters Leys, Bishop's Stortford, Herts, CM23 2RH. **A**

ARCHIMEDES BUFFER PODULE

This single width podule allows you to extend a fully buffered podule bus outside the machine. Connect 8 or full 16 bit simple or external podules, to a 1 metre 64 way ribbon cable. This means you can :-

- ◆ Develop new podules on the bench.
- ◆ Test/ examine/ analyse/ modify/ existing podules.
- ◆ Connect to an external box to increase available podule space.
- ◆ Demonstrate podules in schools and colleges.
- ◆ Buffer chips are socketed and are easily replaced.
- ◆ Quick and easy modification to podules.
- ◆ External podule box will be available in July.

In stock now and despatched the same day on receipt of your cheque / postal order for £49-50 inclusive.

SGB COMPUTER SERVICES

140 Disraeli Rd, London SW15 2DX. Details of products on request.

TEL: (01) 874 5675

Assembly Language Programming

Alan Glover

When using a computer, it is easy to be unaware of what goes on behind the application you are running. The application may perhaps be a BASIC program, but BASIC itself is a program, so what language is that running in?

The answer is ARM machine code – the instructions which make the central processor in the computer perform small tasks. Programming in machine code, using assembly language, has some advantages over using BASIC or other applications; the program usually runs faster and it can interact much more efficiently with Arthur.

The terms ‘machine code’ and ‘assembly language’ may be unfamiliar to you but to explain them we must first take a look at what goes on inside a computer, or more precisely inside its central processing unit (CPU). The ARM itself is different from most commonly used processors, because of its use of RISC technology. For simplicity, we will consider first a conventional CPU and then modify the ideas to reflect the ARM’s improvements.

Three groups of wires, called busses, are connected to the CPU. Each of these wires can be at one of two states, usually known as 1 or 0. Hence, the number of different combinations is $2^{(\text{number of wires})}$. The first bus is called the Address Bus, which is used to identify which location in memory the CPU wants to write to or read from. The Data Bus is then used to carry data to or from that location. The last bus, the Control Bus, performs various functions including indicating to the rest of the hardware whether the CPU wants to read or write data.

The data carried on the Data Bus can be just data but more often it is an instruction code telling the CPU to do something. Inside the CPU are a number of special storage locations called registers which are used for internal operations, an Arithmetic and Logic unit which performs arithmetical and logical operations (surprise!) and other controlling logic.

These instruction codes, read from memory locations using the data bus, are collectively called Machine Code.

Most CPUs work in three distinct phases. In the Fetch phase the next instruction to be performed is

collected from memory. The Decode phase lets the CPU decide what it has to do. Finally in the Execute phase it performs the instructions. It then fetches, decodes and executes the next one and so ad infinitum. A weakness of this approach is that only one of these three elements is active at one time.

Obviously the CPU must have some way of telling where to get the instruction from. This is achieved by using a special register called the Program Counter. This holds the address of the instruction to be executed. As with BASIC, execution usually proceeds sequentially, unless some form of branching instruction occurs.

In addition, the CPU must have a way of performing ‘conditional branching’. This is achieved by having another special register called the Status Register. This has several bits, each having a special significance. E.g. one might be a set to 1 if the result of the last sum was zero and set to 0 otherwise, another might be set if interrupts are disabled and so on.

Interrupts are an essential feature of most CPUs. They allow the operating system to do housekeeping regularly and allow peripheral devices which need the CPU to deal with them, to attract the CPU’s attention.

An interrupt to a CPU is analogous to a telephone ringing. You become aware of the request for you to stop what you are doing and answer it. You then decide whether to answer it, or to ignore it and wait for it to go away. If you do answer it you suspend what you are doing whilst talking and usually return to what you were doing on completion of the call continuing from where you had got to.

A feature of the Acorn computers is that they all generate periodic interrupts to the CPU, during which it performs ‘invisible functions’ like reading the keyboard, running the sound system, controlling the flow of data to a printer and so on.

Above, we stated that the groups of bits read in on the data bus for execution were the machine code of the system. However, trying to remember these values to write programs is unnecessary effort. Instead, the manufacturer of the CPU will have defined an Assembly Language for the CPU. This

consists of short groups of letters, called mnemonics, chosen to indicate the purpose of the instruction. The binary code 00011000, which (for example) causes the CPU to clear a flag called 'a' could be given the assembly language mnemonic CLA – representing CLear flag A.

The Acorn Risc Machine

The ARM is a RISC (Reduced Instruction Set Computer) CPU. This term describes the design philosophy of the CPU – essentially that instructions are kept short and execute fast.

The ARM uses 4 byte instructions. This means that there are 32 bits available to encode both the operation required and its parameters. Unlike some CPUs all data required to execute the instruction is present in the one instruction, so there is never a need to fetch operands from memory. The address bus is 32 bits wide (well, that's simplifying things a bit) making it a '32 bit processor'.

Inside the ARM there are 16 registers visible to the user at any one time, called R0-R15. R15 combines the functions of Program Counter and Status Register as defined above. All the others may be used for any purpose, though some have uses defined for them within Arthur and R14 has a special use associated with one instruction. (In fact there are more than 16 registers because the ARM can work in four different modes and alternative copies of some registers are used in the other modes.)

The ARM executes instructions very rapidly using a technique called 'Pipelining'. This consists of fetching the first instruction, then decoding it, but also fetching the second instruction. Finally the first instruction is executed, meanwhile the second is being decoded and the third fetched. This means that instead of the fetch, decode and execute sections being idle two-thirds of the time they are all used continuously.

Next month we will look at our first assembly language program and explain more about the construction of each 32 bit instruction.

(You will need two books to follow this series and write assembly language yourself. The first is Acorn's Programmers' Reference Guide which details how to interface with Arthur, i.e. what calls to the system exist and how to use them. The other is a book about Assembly Language Programming. Until now only one book has been available (Assembly Language Programming by Peter Cockerell, published by MTC) but it was written with the ARM evaluation system in mind rather than the Archimedes, so makes no mention of Arthur. DABS Press are publishing their own work on the subject in June, which I hope to use as the backup reference to this series – I'll give you more details when I've seen a copy.) **A**

NewsMaster – DTP Package

David Haynes

NewsMaster is the first piece of MS-DOS software that I have come across which is specifically released for the Archimedes. The packaging is labelled "RM Nimbus & Acorn Archimedes version". As the name suggests, NewsMaster is of the DTP school of software allowing the user to produce things such as newsletters with facilities to edit and re-size text, pictures and headlines.

The software is supplied on two 3.5" discs, one containing the program(s) and fonts, the other with a library of "clip-art" and numerous printer drivers. The program can be run on a machine with a single

drive or twin drives or it can be installed onto hard disc using the install program included.

Although released for the Archimedes 310, there is no mouse driver, so control is via keyboard only, with selections from the icon menus being made with the function keys. The icons are clear and generally self explanatory. I liked the HELP facility – pressing Alt + function key produces a one line description of the action of the function key.

Page layout is very simple to control. The default setting is for a headline area and the remainder of the A4 page divided into 2 columns. The number of

columns can be increased up to 10 or, if preferred, the page may not be divided into columns at all. The separate headline area can also be removed. Each page of a multi-page newsletter can have a different layout.

Text can be entered from the keyboard or imported from a wordprocessor if previously saved as an ASCII file. A good variety of font styles and sizes is available – 8 to 60 point, normal, bold and italic. Some fancy fonts, Scribe (Olde English), Script, Hampton, Twig and others add extra variety. Once text has been entered and the effect seen, it is a simple matter to mark a block of text and re-size or alter font. Left, right, centred and fully justified text is supported. Simple wordprocessor functions such as copy or cut-and-paste are available.

Over 100 pictures are included in NewsMaster and if these are not enough, more can be imported directly from PrintMaster and similar packages. Artwork can be positioned and then re-sized up to the full A4 size if required. Pictures can be manipulated in a variety of ways; flipped horizontally or vertically, cropped or stretched. I found the UNDO function useful here! Text can be made to flow over or round artwork. Lines and boxes (clear or pattern-filled) can be drawn.

The 68 page manual is good. In the tutorial section, the user is lead through the production of a two-page newsletter, using text supplied. Being very impatient, I can never follow such tutorials and quickly decided to branch out and “do my own thing.” Here was where NewsMaster really proved its worth. The program feels right and works just as one would imagine it should. I only needed to refer to the manual occasionally to check if I could do something.

Inevitably there are some niggles. Using PC-Emulator version 1.09 the program is rather slow, especially if using large fonts, since the screen redrawing process seems to take forever. I found that changing to full page view helped here – “greeked” letters may be unintelligible but at least I could type at full speed and then go back to normal

view to check spelling! Using imported text would also help to speed things up. The manual states that some 90-odd printers are supported. I have a battered old Mannesman Tally MT80 and although listed in the manual, it is not present in the configuration list. I tried many different “similar” printer settings, none of which produced satisfactory results – an A4 sheet being compressed into about 2/3 of a page. Eventually I resorted to PC-Tools to modify the printer file (altered the n/216th inch settings). Now, with the modified HP Think Jet printer selected I obtain very good results! All the “standard” printers ARE supported; Epson, Citizen, IBM, NEC, Panasonic, Tandy, etc., so there should not be too much trouble there. I have tried Epson FX and MX and they worked perfectly.

There are many Desk Top Publishing packages around aimed at a similar market. I have worked with Aldus Pagemaker on IBM compatibles and clearly NewsMaster is much less sophisticated, but there again it is very much cheaper and easier to use. When compared to the horrors of Fleet Street Editor on a BBC micro or even AMX Pagemaker, NewsMaster is definitely a joy to use. Being a school teacher, I have many “guinea pigs” eager to try out the latest piece of software in which “Sir” has invested his hard-earned cash. Having temporarily lost the manual, I left one group to produce a news-sheet with no help at all, other than a scribbled note explaining how to load PC-Emulator, MS-DOS and then NewsMaster. The fact that they had produced a single page news-sheet with graphics – albeit rather critical of “Sir’s” atrocious memory, within an hour, speaks volumes for the effectiveness of the NewsMaster package.

NewsMaster

Produced by Kyocera Unison, Inc. U.S.A.

UK Distribution by LTS Ltd, Haydon House, Alcester Road, Studley, Warks, B80 7AN. (0386-792617)

Price £60 +VAT **A**

Sound Synthesis – Part 3

Ian Nicholls

The first article in the series gave brief introduction to the concept of wavetables and how the Archimedes can generate sounds using them. It also contained parts of an original Acorn program called FMBass14 with which you could create a new voice sounding something like a trumpet. The second article concentrated on two programs, called FileGen and Arc_Wave, which enable you to create an almost infinite variety of new voices using harmonic, simple FM and complex FX synthesis. In this article I want to explore a few aspects of the wavetables created by Arc_Wave, to give you the missing parts of the original Acorn program, FMBass14 (which imply hidden extra facilities in WaveSynth) and to suggest some further developments of last month's programs which other Archive readers might like to have a try at.

Arc_Wave Wavetables

Probably the first thing that will have struck you about the wavetables created by Arc_Wave is their size – very long! You will soon run out of space to hold them, even on an 800k disc! If you have studied the data that is input to FileGen, you may have realised that most of the separate segments in each wavetable are identical. As an example, let us have a look at the data for the "Organ" voice from last month's article; the amplitude breakpoints for the five harmonics are as follows:

1		2		3	
time	amp	time	amp	time	amp
0	0	0	0	0	0
3	1.0	3	0.7	3	0.4
6	0.8	6	0.55	6	0.3
250	0.8	250	0.55	250	0.3
300	0	300	0	300	0
0	0	0	0	0	0

4		5	
time	amp	time	amp
0	0	0	0
3	0.2	3	0.1
6	0.15	6	0.08
250	0.15	250	0.08
300	0	300	0
0	0	0	0

For each harmonic, its value does not change from the sixth segment to the 250th (i.e. from 6 centiseconds to 2.5 seconds: each segment lasts a centisecond). The result is that the overall loudness and waveform shape of an "organ" note do not alter from 0.06 second after it starts until 2.5 seconds later. It seems a bit of a waste of space to store the same information in 245 separate segments (numbers 6 to 250 inclusive). It is done because we might need the note in a piece of music for two or three seconds, e.g. a semibreve. The note should maintain the same volume and "timbre" (mix of harmonics) whilst it is sounding. To achieve this, WaveSynth needs to be given the same data as it moves through successive segments.

Looping

Ideally, we need a facility within WaveSynth which would enable it to loop – to continue to output the same segment, or series of segments, until a note is released. It would then move to the segment at the start of the release phase. When a musician stops playing a note on an instrument, an organ say, the sound from the note does not necessarily stop immediately – that depends on the instrument and the acoustics of the place where the instrument is being played. In a church, it may take 3 or 4 seconds for an organ note to die away.

This part of the sounding of a note, the more-or-less dying away to nothing, is called the "release" phase. WaveSynth jumps to the "endseg%" segment at the start of the release phase of a note. However, the sound will immediately be cut off if a new note is played on the same voice. Now, as you will see below, there may well be such a looping feature in WaveSynth – it just awaits discovery! If a looping facility were available, the length of many wavetables could be reduced by at least 75%!

Clicks Between Notes

With some of the voices, you may notice a "click" or percussive noise between notes. It is possible that some experimentation of the amplitude envelopes and judicious choice of release segments may reduce this. It is possible that the use of a cosine wave rather than sine wave in "Arc_Wave" may help.

FMBraSS14 – the Missing Lines

If you followed the line numbers that were given with FMBraSS14, the missing line numbers will be inserted in the right places. Lines 690 to 1140, 1240 and 1250 test to see whether the amplitude and modulation values change from one segment to another. If they do not, a wavetable pointer (ptr%) is decremented by 256 (the number of values in a segment.) The program also tells you the number of such segments in line 1240. In line 1250, the program prints out the number of the first “zero-change” segment and calls it the “sustain” or “repeat” segment (repeatseg%).

Now all of this is exactly what you were looking for above in the section on “looping”. The only problem is that, having calculated the value of repeatseg% the program does nothing with it! There would seem little purpose in going to the trouble of working out whether each segment is compactable and the start of the sustain phase of the wavetable, without WaveSynth being able to make some use of it. The challenge for you is to find out how to make this assumed feature of WaveSynth work!

Plotting the Segments

The new line 1190 calls a procedure named “PROCplot”: no prizes for guessing what this does. In fact, it will plot the shape of the calculated waveform for each segment. However, it will only do this if you hold down a key – line 1390 tests for a key being pressed. Line 1410 is an alternative to line 1400 which gives you a more compact plot.

```

690 ptr%=WaveTable%
700 :
710 flat%=FALSE
720 compact%=FALSE
760 repeatseg%=0
840 IF dA<>0 THEN flat%=FALSE
920 IF dA<>0 THEN flat%=FALSE
1020 IF flat% THEN
1030 ptr%=ptr%-256
1040 PRINTTAB(0,2);"Compactable
    segment: ";t%DIV256;" "
1050 compact%=TRUE
1060 ELSE
1070 flat%=(dA=0) AND (dI=0))
1080 IF flat% THEN
1090 PRINTTAB(0,2);"Zero-change
    segment: ";t%DIV 256;" "
```

```

1100 repeatseg% = t% DIV 256
1110 ELSE
1120 compact%=FALSE
1130 ENDIF
1140 ENDIF
1190 PROCPlot(t%,l%)
1240 PRINT (WaveTable%-ptr%)DIV256;
    " segments compactable"
1250 PRINT "Sustain (repeat)
    segment: ";repeatseg%
1350 :
1360 DEF PROCPlot(t%,p%)
1370 LOCAL x%,y%
1380 IF t%=0 THEN CLS:PRINTTAB(0,0)
    ;EndTime/256;" segments"
1390 IF INKEY$(0)="" THEN ENDPROC
1400 x%=2*(t%MOD512)+128
1410 REM x%=.25*(t%MOD4096)+128
1420 IF (t%MOD256)=0 THEN PRINTTAB
    (0,1);t%/256
1430 IF compact% THEN ENDPROC
1440 y%=512
1450 MOVE x%,y%+256:PLOT 7,x%,y%-256
1460 MOVE x%,y%
1470 y%=p%>>1
1480 IF (p%AND1)=1 THEN y%=-y%
1490 DRAW x%,512+2*y%
1500 ENDPROC
1800 STOP
```

The BEEP WaveTable Data File

Together with the program FMBraSS14, Acorn responded to my original enquiry about WaveSynth with the listing below of the data file that was used to create the wavetable for the “Beep” voice that WaveSynth normally produces. The listing includes the set of values for the single segment that the “Beep” consists of. Whereas the FMBraSS14 program reads a set of amplitude and modulation breakpoints which are used to generate a whole series of wavetable segments, “Beep” consists of just one wavetable segment and an amplitude envelope. Obviously, WaveSynth is capable of working with at least two different ways of creating sounds from wavetable and amplitude envelope data. We need to play the ace detective again to work out what parameters have to be fed to WaveSynth in the wavetable header fields to activate these different features!

The text after the semicolons consists of comments entered by Acorn. Like you, I can only guess what

the comments mean. By comparing the format of the listing with the disassembled WaveSynth module, it is evident that DCD is equivalent to the basic assembler EQU and that the construct &0000007F + 1:SHL:9 takes the value 1, shifts it left by 9 places and adds it to &0000007F to give the value of &0000027F.

```
WaveTable0
;header
= "!WT:" ; magic ID word
WaveNameStart
= "Beep"
= 0,0,0,0
= 0,0,0,0 ; room for MAX 11 - char
name + 0C terminator
WaveLen * . - WaveTable0
DCD End_Of_WaveTable -
WaveTable0 ; total length
WaveStart * . - WaveTable0
DCD 8,8,8,8,8,8,8,8 ;eight pitch
related entries!
WaveEnd * . - WaveTable0
DCD 13 ; end ptr
DCD 0,0
; end of 16 words header...

; offset 64 (index 8)
; descriptor 8 (ATTACK)
DCD &0000007F + 1:SHL:9
DCD &00090001
; descriptor 9 (DECAY)
DCD &000000F0 + 31:SHL:9
DCD &000A0001
; descriptor 10 (SUS a)
DCD &00000080 + 500:SHL:9
DCD &000E0001
; descriptor 11 (SUS b)
DCD &000000DF + 25:SHL:9
DCD &000A0001
; descriptor 12 (SUSTAIN)
DCD &00000000 + &FFFFFF:SHL:9
DCD &000D0002
; descriptor 13 (release)
; descriptor 9 (DECAY)
DCD &00000080 + 1:SHL:9
DCD &000E0001
; descriptor 14 (Dead)
DCD 0
DCD 0
% 256 - (. - WaveTable0)
; REM WaveGen output ; One WaveTable
data segment
```

```
DCB &40,&68,&80,&8C,&9A,&A2,&A8,&AE,
&B6,&BC,&C0,&C4,&C6,&CA,&CC,&D0
DCB &D2,&D4,&D8,&DA,&DE,&E0,&E0,&E2,
&E4,&E4,&E6,&E8,&E8,&EA,&EA,&EC
DCB &EE,&EE,&F0,&F0,&F2,&F2,&F4,&F4,
&F4,&F6,&F6,&F8,&F8,&FA,&FA,&FA
DCB &FA,&FC,&FC,&FC,&FC,&FE,&FE,&FE,
&FE,&FE,&FE,&FE,&FE,&FE,&FE,&FE
DCB &FE,&FE,&FE,&FE,&FE,&FE,&FE,&FE,
&FE,&FE,&FE,&FC,&FC,&FC,&FC,&FA
DCB &FA,&FA,&F8,&F8,&F8,&F6,&F6,&F4,
&F4,&F4,&F2,&F2,&F0,&F0,&EE,&EE
DCB &EC,&EA,&EA,&E8,&E8,&E6,&E4,&E4,
&E2,&E0,&E0,&DE,&DA,&D8,&D4,&D2
DCB &D0,&CC,&CA,&C6,&C4,&C0,&BC,&B6,
&AE,&A8,&A2,&9A,&8C,&80,&68,&40
DCB &41,&69,&81,&8D,&9B,&A3,&A9,&AF,
&B7,&BD,&C1,&C5,&C7,&CB,&CD,&D1
DCB &D3,&D5,&D9,&DB,&DF,&E1,&E1,&E3,
&E5,&E5,&E7,&E9,&E9,&EB,&EB,&ED
DCB &EF,&EF,&F1,&F1,&F3,&F3,&F5,&F5,
&F5,&F7,&F7,&F9,&F9,&F9,&FB,&FB
DCB &FB,&FD,&FD,&FD,&FD,&FF,&FF,&FF,
&FF,&FF,&FF,&FF,&FF,&FF,&FF,&FF
DCB &FE,&FF,&FF,&FF,&FF,&FF,&FF,&FF,
&FF,&FF,&FF,&FD,&FD,&FD,&FD,&FB
DCB &FB,&FB,&F9,&F9,&F9,&F7,&F7,&F5,
&F5,&F5,&F3,&F3,&F1,&F1,&EF,&EF
DCB &ED,&EB,&EB,&E9,&E9,&E7,&E5,&E5,
&E3,&E1,&E1,&DF,&DB,&D9,&D5,&D3
DCB &D1,&CD,&CB,&C7,&C5,&C1,&BD,&B7,
&AF,&A9,&A3,&9B,&8D,&81,&69,&41
End_Of_WaveTable
```

Some Further Possible Developments

I will leave you with a few ideas of ways in which FileGen and Arc_Wave could be developed: At the moment, I do not have time to do any of this work myself! Anyway, why should I have all the fun?!

Rather than require users to work out the shapes of the amplitude and modulation envelopes and the values to type into FileGen, it would be useful to be able to draw the shapes interactively on the screen with the mouse. An added bonus would be for previously-created data files to be loaded in and displayed in graphical form for modification.

In the first article, I gave you the series of commands needed to load in a new voice created with Arc_Wave. To do this it is, at present, necessary to start from the BASIC prompt, enter a SYS"OS_Module" command and then QUIT and

enter a further RMREINIT command from the Arthur prompt. It would be much more convenient to be able to do this at the end of Arc_Wave or for that program to call an assembly language routine to ask you which voices you want to load.

On the subject of assembly language, there is a real potential here for speeding up the wavetable creation part of Arc_Wave. It would also be most helpful to be able to listen to the new voice as soon as it has been created. (Not surprisingly the above features are all available in the only commercial competitor to our two programs – EMR's SoundSynth, reviewed in May Archive.)

Creating New Voices

You may well have wondered where I obtained the data for the example voices in the second article. The clarinet and trumpet data was measured by eye from graphs in the second of two references by Hal Chamberlain, listed at the end of the first article. Chamberlain's reference source was a sequence of articles under the general heading of "Lexicon of Analysed Tones" in the "Computer Music Journal", published during 1977 and 1978. The Brass14 data was copied from the original Acorn article; the piano data was contained in an article by Bill Schottstaedt called "The Simulation of Natural Instrument Tones using Frequency Modulation with a Complex Modulating Wave" in the Roads and Strawn reference from the first article. The Organ data was just guesswork on my part!

Well, that's the end of this short series. It is now up to you to experiment with FileGen and Arc_Wave. Please let us know of any interesting or unusual voices that you have created with them and list the necessary data for FileGen. If any of you do pick up the challenge of further exploration of WaveSynth, happy detection and please share your deductions with the rest of us.

Comments on the EMR SoundSynth

I have now been tempted by the opposition and have bought a copy of EMR's SoundSynth program. It is an extremely good package but the documentation that comes with it is, in my view, completely inadequate. The novice Archimedes user will not be able to make any sense of it, since it assumes a

familiarity with the concepts of waveforms, harmonics, etc. which I have tried to introduce in my series of articles. Also, the package is capable of many things which the documentation and even the on-screen prompts just do not explain properly. I am convinced that there is real potential for a book to be written that fully explores the facilities that SoundSynth opens up on the Archimedes!

Two particular features that the slim manual does not cover are FM synthesis and using the "Welcome" disc Music Editor to play the sampled sounds on the SoundSynth disc and on its companion "Creations". I accept that the manual does make reference to playing your own sounds with the Music Editor, but if you try to play the sampled sounds they are completely out of tune. The reason for this is that the sampled sounds were recorded at a variety of different pitches, none of which is the same as the pitch used for sounds that you create yourself with SoundSynth.

SoundSynth Tuning

I have been engaging in some more detective work, with SoundSynth this time, and the pitches at which the various sampled sounds were recorded seem to be as follows (using the octave numbering convention in the User Guide):

SoundSynth Disc

Bass Guitar	E4	ChurchBell	C5	Laugh	E5
OrchHit	E6	Piano	C5	Sax	G4

Creations Disc

AcGuitar	C5	AcBass	C3	Atlantis	C3
Blub	C5	Choir1	A#5	ChurOrgan	C4
ElecTom1	E4	EndIt	C4	FunnyVox	C5
Harpl	C5	Harpsi	C5	HiString	C6
HonkyPiano	C5	JapFlute	C6	JazzOrgan	C5
MidStrings	C5	NoteArp1	D6	Opera	E6
OrchChoir	E4	OrganHit	C4	Timpani	E3
TomTom1	E4	Trombone1	C5	Trumpet	C6
Vibe	E4	Whistle	C7	Xylophone	C6

In the User Guide convention C5 is middle C. To make middle C sound at the right pitch for voices such as Trombone1 which were recorded at middle C, the Archimedes tuning needs to be altered by using the revised tuning command given in the Addendum to the Programmers Reference Manual. If the command *TUNING -&1AAA is entered after loading any of the "C5" voices, they will sound

Printkey Screen Dumps (and Saves)

Neil Strong

Wouldn't it be nice to be able, in the middle of some application, to press the Print key and produce a printout of the current screen? Neil shows us how...

Basically, the program intercepts all calls by the operating system to insert a character into the keyboard buffer. Each character (i.e. keypress) is checked to see if it is a CHR\$(128), which is the default for the print key.

This is a bit of a cheat; if the print key is redefined, the program will not work. However, the emphasis here is on simplicity.

If the test detects CHR(128) sent to the keyboard buffer, an OSCLI "HARDCOPYRX" is performed (Or the relevant dump command for your printer).

I have tested the dump from within 'running' basic programs and a number of machine code demos, with no apparent complaints from ARTHUR at being interrupted whilst inserting characters into buffers!

To Install the program

1. Type in the BASIC listing
2. Save the BASIC program before going any further.
3. RUN it – This will save a MODULE file called 'PKEY' on the disk.
4. To install the MODULE, type '*PKEY'

This will install the actual dump code as a module, and will remain resident until a cold reset.

Now, when you press the 'PRINT' key, the current program will be suspended while a screen dump is sent to the printer.

(You can also replace HARDCOPYFX with SCREENSAVE "\$.PIC" and save the current screen to disc – it works, we've tried it. Ed.)

```
10 REM > PRINTKEY
30 REM PrintKey Version 1.00
60 REM (c) 5th May 1988
70 REM By Neil Strong
80 REM
90 REM Intercepts the INSV 'Insert
    Character into Buffer' Vector
100 REM checking to see if the
    PRINT key has been pressed.
120 REM If it has, a *HARDCOPYRX
    command is performed
130 REM
140 REM This can be replaced with
    the printer dump command
150 REM relevant to your printer,
    with parameters if required.
180
190 pc=15:link=14:sp=13
210 DIM code% 500
230 FOR K%=0 TO 2 STEP 2
240 P%=code%
250 [OPTK%
260 .start
270 .module_start
280 equd 0 ; No Start code
290 equd init-module_start
    ; Initialise code
300 equd final-module_start
    ; finalise code
310 equd 0
```

at the correct pitch, but the System ROM voices such as "String Soft" will be unpleasantly off key! The "CS" sampled voices can then be used within the Music Editor. Clearly, some more development work is necessary on my part to shorten or stretch the wavetables for the voices not recorded at middle C, so that they sound as if they had been!

FM Synthesis

FM synthesis can be achieved by altering the waveform used for harmonic synthesis (having

chosen "HARMONICS" from the main screen, press the left-hand mouse button on the "alter equation" box). The default equation for this waveform is $127 * \text{SINRAD}(X)$. If this is changed to $127 * \text{SINRAD}(X + i * \text{SINRAD}(n * X))$ and only the fundamental (i.e. first harmonic) is used, then this will give Simple FM synthesis, with i as the modulation index and n as the modulator to carrier frequency ratio. **A**

```

320 equd title-module_start
           ; *MODULES text
330 equd help-module_start
           ; *HELP MODULES text
340 equd 0
350 equd 0
360 equd 0
370 equd 0
380 equd 0
390
400 .title
410 equs "PrintKey":EQUB 0:ALIGN
420
430 .help
440 equs "Print Key":EQUB 9:equs
    "1.00 (05 May 1988)":equb 0:align
470 ; Initialise code, called when
    module is first loaded, and
480 ; after a RMREINIT or RMTIDY
490 ; Claims the INSV vector (&14)
510
520 .init
530 stmfdd (sp)!,{link} ; save return
                        address
540 adr r1,check ; point to keypress
                        check code
550 mov r2,#0
560 mov r0,#&14
570 swi "OS_Claim" ;claim INSV
                        vector
580 ldmfdd (sp)!,{pc} ; return
590
610 ; This code is called when a
    RMKILL or RMREINIT is performed
620 ; It releases the vector claimed
    on module initialisation
640
650 .final
660 stmfdd (sp)!,{link} ; save return
                        address
670 adr r1,check
680 mov r0,#&14
690 mov r2,#0
700 swi "OS_Release" ; release INSV
                        vector
710 ldmfdd (sp)!,{pc}
720
740 ; This is the functional part,
    called each time a request to
750 ; insert a character into a
    buffer is made.
760 ; I.e. Keypress, RS232 output,
    Printer output.

```

```

770 ; If the print key is detected
    (default 128) a *HARDCOPYRX is
    performed
800 .check
810 stmfdd (sp)!,{link} ; save
                        return address
820 cmp r1,#0 ; check for KEYBOARD
                        input
830 ldmnefdd (sp)!,{pc} ; if not,
                        then pass on call
840 cmp r0,#128 ; is it 'print' key
850 ldmnefdd (sp)!,{pc} ; if not,
                        pass call on
860 adr r0,command ; otherwise
                        point to command
870 swi "OS_CLI" ; execute it
880 ldmfdd (sp)!,{r2} ;pull previous
                        return address off stack
890 ldmfdd (sp)!,{pc} ; claim call
900
910 .command
920 equs "HARDCOPYRX" ; place print
                        command here
930 EQUB 0 ; null terminator
940 ALIGN
950 .end
960 ]
970 NEXT
980
990 OSCLI "SAVE PKEY "+STR$~(start)
    +" "+STR$~(end-start)
1000 OSCLI "STAMP PKEY" : REM Date
    stamp object file
1010 OSCLI "SETTYPE PKEY &FFA" : REM
    set type to MODULE A

```

More Software Available

- **Typefaces for 24-pin printers** from Ian Copestake Software at £19 + VAT include Shadow, Personal, Irish Gaelic and German Fraktur.
- **First Fonts (£19 + VAT)** – Ian Copestake Software have also provided a set of special characters and a facility for allowing on-screen and printer, justified, proportional spacing and typeface selection with First Word Plus. **A**

Games Galore!

Adrian Look

Four new games have arrived on the scene. Hoverbod (£14.95) and Missile Control (£14.95) from Minerva, SAEC from Fairhurst Instruments (£12.95) and Quazer (£8.95) from an independent programmer – Julian Rockey.

HoverBod (and the Ibbly Squiblies)

This game was supplied in a very high quality, yellow library case and cover; inside the disc and instructions. On BOOTing the disc, the Archimedes was re-configured and the disc drive made a faint whirring/crunching noise as the game loaded, which indicated the software protection that Minerva employs – this slows down the loading process considerably and was most infuriating. The auto-configuration did not take into account the fact that some users (with ROM podules) may have the RFS as a default, so the loading process ‘bombs-out’ leaving the machine in a terrible state.

Hoverbod is a fun, real time ‘adventure’ game and requires a degree of both co-ordination and brains. Hoverbod is a yellow spherical droid from the planet Zingle and, to cut a long story short, you have to control the droid through 8 levels collecting ‘red diamonds’ and avoiding the ibbly squiblies, among other things! Having done this, Hoverbod will be teleported to ‘the castle’ where he has to collect no less than 500 feathers in order to build two giant wings to fly home with. No mean feat I assure you!

However, things aren’t all that simple. At the beginning of each level Hoverbod can only move left and right. You have to help him collect thrusters to allow him to jump and guns to allow him to shoot the ibbly squiblies, as well as making sure he has enough oxygen to survive. There are also special functions to turn certain ‘nasties’ on and off.

All in all, the game is very entertaining although the mid levels are a bit easy (Not for me, they’re not! Ed.), which means the persistent games player may find that the game does not have a long life span. However, the graphics were of a good standard and it did give an occasional squeak (when you died).

There is no scoring system, you are only concerned with advancing through the game. Here, a very

convenient default system is implemented. You don’t have to start from level one every time, only the most recent screen (others can be accessed via passwords – see Hints & Tips, page 8!)

An enjoyable game and definitely recommended.

(Parental comment: My two boys are well and truly hooked on Hoverbod, but I don’t mind – it means I can use it as a bribe – “You can play Hoverbod when you’ve laid the table” etc. Also, it doesn’t just depend on manual dexterity – they do have to do some logical reasoning, so it does have a vaguely education slant to it. Full marks from me. Ed.)

Missile Control

Missile Control came in the same type of case as Hoverbod and !BOOTed up in exactly the same way. Having booted up, that is where the similarity ends. The intermediate loading screen has a rotating cylinder with the Minerva Logo on it – this was quite impressive. There was also some sampled sound: “Commander to Missile Control – Attack warning. Prepare defensives.” (We had to use headphones to work out what it said – it was incomprehensible through the internal speaker.)

I am sure that the game needs no introduction, as it is a very long established arcade game converted for the Archimedes. The graphics and sound are in true Missile Control style – and of a good standard! However, at the end of the game there is an extremely loud crashing noise and a flashing white screen which is totally unnecessary, irrelevant to the game, and really rather unpleasant.

The game does tend to slow down when there is a lot of action going on; and when all the villages are destroyed the game continues until the ‘attack’ was finished – which can be quite a while.

Generally, the game was well done and if you enjoyed ‘Missile Command’ on any other machine, the Archimedes version is recommended.

SAEC

This is another ‘converted’ game. The theme is centred around a ‘bomb disposal’, bug named Boris. He can jump between 5 levels, and on each there is a separate bomb. Boris must make sure that

none of these blows up or he will 'kop-it'. This is not all – there is also a 'nasty' which tracks Boris and if you don't watch out it will jump out on you quite unexpectedly – seems like a good idea for an action-packed arcade game.

Having said this, I really didn't think it was anything more than a re-hashed BBC game. The graphics and sound are certainly different, but I am afraid not to my liking. Personally, I thought the BBC version – even if SAEC is in stereo sound.

I reckon it is over-priced at £12.95 – both the Archimedes and Nigel Stuart are capable of better!

Quazer

This last game is a classic action-packed shoot-em game. The graphics are very smooth, with a

scrolling background and hordes of aliens descending as you try to penetrate their defences and blow-up your objectives. There are eleven levels each getting progressively harder (they get sufficiently hard to keep you amused for hours).

The controls allowed you to move up & down as well as the usual left & right. The fire button is 'auto-repeat' which saves your keyboard.

The game was fun to play. If you enjoy shoot-em games, Quazer will have you hooked and at £8.95 it is good value!

(If you get fed up and want to cheat, you can try the machine code patch given in Hints and Tips, page 8. Also, the Passwords are there if you want to use them.) **A**

Euclid – For High Speed 3D Pictures

Brian O'Carroll

Euclid is a three dimensional 'object processor' system from Ace Computing, specially written for the Archimedes. It comes as a relocatable module which forms the heart of the system and a suite of programs written in BASIC.

The program which boots up from the disk is the Euclid design program. It allows you to design objects by using any of 6 plane views (Front, Left, Above etc.). You can also watch your designs grow in a 'View' window. The pictures may be drawn as a wire frame or with full hidden line removal, no matter what shape the objects are. The View window starts as a plane isometric view with hidden line removal and face colouring. The 'design windows' begin as wire frame. You can create planes, surfaces and tube-like structures. The general idea is that you plonk on your rough shape in the form of polygons and rectangles and the program will join the dots, fill in the planes and display the results in all the windows.

Euclid uses a hierarchical system to store shapes in memory (or in files). It has three main categories called Objects, Solids and Landscapes. Solids and Landscapes are like simple files on a disk and contain the actual data for doing the drawing. Objects are like directories and contain other Objects, Solids or Landscapes. A Solid is a

collection of planes like rectangles, pentagons etc. A Landscape may be a tube made by joining the points on two or more polygons; a box type tube where the polygons are rectangles; or a surface made by joining points on adjacent lines (like the landscape in the lander game).

Once something has been defined, it may be made to appear duplicated somewhere else in the picture. When designing a table you may decide you need four identical legs. You define one of them, put it at one corner, then put copies of it at the other corners. You have only created one leg and if you edit the shape of any one of the legs, all the other images will change too.

Since, as you go along, you name each piece of the scene that you create, you may pick out the name of the Object, Solid or Landscape from a list, to be inserted anywhere in the picture.

The actual screen layout is almost completely WIMP driven. The options come up from pull up menus on the bottom of the screen. An editing menu is also available from any of the design windows, by pressing the centre mouse button.

Menus

The first menu is the disk menu giving LOAD, SAVE, MOUNT, DIR, etc. and an option called APPEND. The APPEND option allows you to add

the definitions of Objects, Solids and Landscapes from another file and insert the images into your design. This provides a library facility, like a procedure library, but made of shapes instead of routines. There are some files provided on the disk, including one of 'Platonic Solids' (Octahedron, Icosahedron, Dodecahedron etc...), which is a useful start to any object library.

Next there is the print menu. This provides access to the built-in dump routines for Epson compatible printers. You can invert the printout, print the whole screen, menus and all, or just what is in the 'View' window – in an isometric view, or any style (wire frame, solid etc..), any size, from any viewing angle, with or without perspective.

The third menu is the 'goodies' menu. This includes QUIT and NEW and options to change the colour of the faces and edges of the objects. It also allows *-commands and has options to save the screen as a sprite (or just the View window). You can change the screen mode – this is useful for printing because, when you change to a two colour mode, all the colours are drawn as patterns (hatching, shading etc) so you can see exactly the pattern that will be dumped.

Finally, the windows menu which is a list of all the possible windows that may be displayed. That includes ABOVE, LEFT, RIGHT, BELOW, FRONT and BACK for the design windows, the VIEW window (also known as CAMERA #0); the CAMERAs numbered 1 to 9; and two windows called CONTROL and CAMERAS.

Cameras

The cameras are quite important. They can provide you with a view from anywhere in your scene. Cameras may also be attached to objects, so that when you move the object, the camera moves with it. The design windows are different from the camera windows since they only display the component that you are working on at the moment. The control window controls the view through the design window and shows the current object, solid, landscape, plane, point, vane etc.. that is being edited and its parent.

For example, if you have a jug (as I did) you would design the handle and the body separately. When designing each, the parent object is either BODY or HANDLE. Once both are done, you go up in the

hierarchy so that the parent object is JUG. At this time both the body and the handle are displayed. You may then scale, rotate and position each individually so that the handle actually becomes part of the jug. The program will attempt to match corners of objects for you if it is obvious that is what you are trying to do.

Scaling, positioning and rotating may be done in one of two ways. Using the mouse in the design windows, you may 'drag' the current component to a position (SELECT button) and drag it to the correct rotation and size (ADJUST button).

This may also be done from the control window by altering the data about the object that is displayed there, like Rotation about X, Y and Z axis, offset in directions X, Y and Z and scaling as a factor of the original definition size.

In the review copy I was using, there were some minor bugs, which have been straightened out for the release copy, which also has additional features such as saving the camera positions to disk; allowing you to swap the Y and Z axis labels throughout the program. *(There is also more printer support now. Ed.)*

Conclusion

I had a good time reviewing Euclid and managed to use most of the system without reading the manual. (I found the first draft of this OK, provided you read it from the start, but less experienced users may need more help, which apparently the release version will give.) It took me a few hours to investigate the examples supplied (including the inevitable goblets, of course) and found it quite easy to produce such things as a house of cards, an anglepoise lamp, a tea chest full of jugs and flower bowls and a sock! My latest idea is to build a TV studio, complete with news-desk and actually attach the cameras to the ones in the studio. It is possible – I've tried it, but I'm waiting for a release copy, so that I can save the camera positions.

Those of you who want to use the definitions in programs will have no trouble. The relocatable module provides the user with new SWI commands (or SYS from within BASIC) which draw the 3D pictures from the data loaded into memory from disk. There are no secrets held about the module and several example programs are included, that are

almost self explanatory. I messed around with a couple (as recommended in the manual) and produced a pencil and house of cards, rotating in opposite directions, in real time, with full hidden line removal and colouring – a 20 line program. It is a very open ended and versatile system, allowing you to choose exactly what you want to do. You may choose any single part of your definition and do with it as you will.

Apart from defining objects using the program and displaying them using BASIC, you can define objects using BASIC (or any other method) to produce a file that may be appended into the design program and edited from there or added to other definitions. This is how the platonic solids were created and the program to do that is included (for you to fiddle around with).

All the SWIs are listed and the file structure of the design definitions is given. There is even a little guidance on programming techniques. There is an expected extension to this system which will allow you to 'record' a sequence and replay them at 50 frames a second! (how many?)

Well I thought it was about time somebody started using the Archimedes as an Archimedes, rather than a fast BBC. Congratulations to Ace computing, who have done just that. In my opinion, Euclid will be a great success and a useful tool for any future 3D design programs (A standard even). It combines versatility with ease of use and even loads quickly!

Available at £45, mail-order, cheque-only from: Ace Computing, 27 Victoria Road, Cambridge, CB4 3BW, (0223) 322559. **A**

SigmaSheet Sorting Program

David Palmer

Being a teacher, I have several hundred pupil names in spreadsheet files. I have always used spreadsheets in preference to databases, as 25 or so pupils can be seen on the screen at once. The up and down arrows also allow easy movement between the same fields of different pupils and updating the latest test mark into the file from a pile of test papers in a random order is therefore relatively easy.

However, I needed a way to sort pupils entered in one order, into another order, perhaps alphabetical, by order of register or subject classes, or by marks. SigSort is a short program to do this. I am sure the principles behind it would be equally applicable, say, to a Viewsheet or Intersheet file, or one created in a wordprocessor. I have already used it to transfer sorted ASCII data to Pipedream.

It reads an ASCII file created by the SigmaSheet "print to disc" option from the main menu, in which the columns are separated by TABs (CHR\$ 9) and the rows are separated by RETURNS (CHR\$ 13). Each row is read in turn into an array and it sorts the rows on the column requested by the user, in ascending or descending order. It was originally designed to sort on strings, which were pupil names, but has now been modified to sort on numbers.

Make all columns of the spreadsheet file wide enough so that no data is hidden before printing the

sheet to disc. Note the size of the sheet in columns and rows before leaving SigmaSheet for BASIC to use the SORT program.

The program shows the catalogue of the current directory and asks for a source filename. It allows *-commands to be put in to let you change directories or discs. *DIR:0 may be useful here, for instance. A destination filename for the sorted ASCII file is then requested. The sorted ASCII file is created, with columns separated by commas and rows by RETURNS. By trial and error, this was found to be the easiest format to import back into SigmaSheet.

Choose <U> for the SigmaSheet import utility from the Banner screen and choose the comma-separated file option to import the sorted file, which you have just named. The new SigmaSheet filename is then selected by the user. <ctrl-Q> (rather than <tab> as suggested) or </> then <escape> gets you out of the import utility, then <shift-break> and choose <S> for SigmaSheet from the Banner screen.

Take care not to have any commas in numbers, or in comments, as these are used to separate the fields or columns and the bit after the comma will be imported into the wrong cell.

This program was written in a hurry to do a specific job and is not the most wonderful example of programming you will ever see. I cannot guarantee

that it is bug free and you will almost certainly be able to crash it if you try! Please accept it for what it is and modify it to your heart's content!

```

10 REM > $.SigSort
20 REM SigmaSheet Sorter
30 REM by D.C.Palmar, 6/6/88
40
50 PROCinitialise
60 PROCopen_read_file
70 FOR R%=1 TO rows
80   PROCread_row(R%)
90   PROCprint_row(R%)
100  PRINT
110 NEXT
120 PROCclose_file
130 PROCsort
140 PROCopen_write_file
150 FOR R%=1 TO rows
160   PROCprint_row(R%)
170 NEXT
180 PROCclose_file
190 PRINT'"File ";destinfile$;
   " sorted and saved to disc."'Now
   use Sigmasheet Import utility,
   choosing comma-separated file
   option."
200 END
210
220 DEFPROCinitialise
230 MODE 12
240 PROCtitle
250 PRINT'"Please give the size of
   the spreadsheet to be sorted."'
260 REPEAT
270   INPUT'"How many columns? "
   columns
280   INPUT'"How many rows? "rows
290 UNTIL columns>0 AND rows>0
300 DIM sheet$(rows+1,columns)
310 PROCwait
320 FOR N%=1 TO 2
330   PROCget_filename(N%)
340 NEXT
350 ENDPROC
360
370 DEFPROCget_filename(number)
380 CLS:*.
390 file$=""
400 REPEAT
410   PRINT'"Enter name of ";
420   IF number=1 THEN PRINT"source";
430   IF number=2 THEN PRINT
   "destination";
440   PRINT" file or STAR COMMAND:"
450   INPUT'string$
460   IF LEFT$(string$,1)="*" THEN
470     command$=RIGHT$(string$,LEN
   (string$)-1)
480   OSCLI command$
490   ELSE file$=string$
500   ENDIF
510   IF number=1 THEN sourcefile$=
   file$
520   IF number=2 THEN destinfile$=
   file$
530 UNTIL LEFT$(string$,1)<>"*"
540 PRINT'"Press SPACE to continue"
550 REPEAT UNTIL GET=32
560 CLS
570 ENDPROC
580
590 DEFPROCopen_read_file
600 OSCLI "EXEC "+sourcefile$
610 ENDPROC
620
630 DEFPROCopen_write_file
640 OSCLI "SPOOL "+destinfile$
650 ENDPROC
660
670 DEFPROCclose_file
680 CLOSE#0
690 ENDPROC
700
710 DEFPROCread_row(r%)
720 sheet$(r%,1)=GET$
730 C%=0
740 REPEAT
750   C%=C%+1
760   REPEAT
770     G$=GET$
780     IF G$<>CHR$9 AND G$<>CHR$13
   THEN sheet$(r%,C%)+=G$
790   UNTIL G$=CHR$9 OR G$=CHR$13
800   slot$=sheet$(r%,C%)
810   WHILE LEFT$(slot$,1)=" "
820     slot$=RIGHT$(slot$,LEN
   (slot$)-1)
830   ENDWHILE
840   sheet$(r%,C%)=slot$
850 UNTIL C%=columns OR G$=CHR$13
860 ENDPROC
870
880 DEFPROCprint_row(r%)
890 FOR C%=1 TO columns-1
900   PRINTsheet$(r%,C%);",";
910 NEXT

```

SigmaSheet Sorter

```

920 PRINT sheet$(r%,columns)
930 ENDPROC
940
950 DEFPROCsort
960 REPEAT
970   PRINT'"ON WHICH COLUMN DO YOU
        WISH TO SORT? - A to Z"
980   INPUT col_to_sort$
990   PRINT'"Sorting on column ";
        col_to_sort$
1000  col_to_sort=ASC(col_to_sort$)
        -64
1010 UNTIL col_to_sort>0 AND
        col_to_sort<=columns
1020 REPEAT
1030   PRINT'"Ascending or
        Descending Sort (A or D)?"
1040   INPUT order$
1050 UNTIL INSTR("AaDd",order$)>0
1060 REPEAT
1070   PRINT'"Numerical or
        Alphabetical Sort (N or A)?"
1080   INPUT sorttype$
1090 UNTIL INSTR("AaNn",sorttype$)>0
1100 PRINT'"Please wait"
1110 FOR current_row=1 TO rows
1120   PRINT(current_row/rows*100)
        DIV 10;"0% sorted":VDU11
1130   FOR R%=current_row+1 TO rows
1140     IF order$="A" OR order$="a"
        THEN PROCascending_sort
        ELSE PROCdescending_sort
1150   NEXT
1160 NEXT
1170 ENDPROC
1180
1190 DEFPROCascending_sort
1200 IF sorttype$="A" OR sorttype$=
    "a" THEN IF sheet$(current_row,
        col_to_sort) >sheet$(R%,
        col_to_sort) THEN PROCswap
1210 IF sorttype$="N" OR sorttype$=
    "n" THEN IFVAL(sheet$(current_row,
        col_to_sort))>VAL(sheet$(R%,
        col_to_sort)) THEN PROCswap
1220 ENDPROC
1230
1240 DEFPROCdescending_sort
1250 IF sorttype$="A" OR sorttype$=
    "a" THEN IF sheet$(current_row,
        col_to_sort)<sheet$(R%,
        col_to_sort) THEN PROCswap
1260 IF sorttype$="N" OR sorttype$=
    "n" THEN IF VAL(sheet$(current_row,
        col_to_sort))<VAL(sheet$(R%,
        col_to_sort)) THEN PROCswap
1270 ENDPROC
1280
1290 DEFPROCswap
1300 FOR C%=1 TO columns
1310   SWAP sheet$(current_row,C%),
        sheet$(R%,C%)
1320 NEXT
1330 ENDPROC
1340
1350 DEFPROCwait
1360 PRINT'"PRESS SPACEBAR TO
        CONTINUE"
1370 REPEATUNTILGET=32
1380 ENDPROC
1390
1400 DEFPROCtitle
1410 CLS
1420 PRINT'"SPREADSHEET ASCII FILE
        SORTER"
1430 PRINT'"WHAT THIS PROGRAM DOES : "
1440 PRINT'"This program reads an
        ascii file created by Sigmashet in
        which the columns are separated by
        TABs (CHR$ 9) and the rows are
        separated by RETURNS (CHR$ 13)."
1450 PRINT'"It reads each row into
        an array and sorts the rows on the
        column requested by the user."
1460 PRINT'"A sorted ascii file is
        created, with columns separated by
        commas. The new filename is
        selected by the user."
1470 PRINT'"After this program has
        made a comma-separated sorted file
        for you, load it into Sigmashet
        using the Import utility and choose
        the comma-separated file option."
1480 PRINT'"BEFORE USING THE PROGRAM:"
1490 PRINT'"    1) Note which column
        you wish to sort on and the number
        of columns and rows in the sheet."
1500 PRINT'"    2) Make all the
        columns of the spreadsheet file
        wide enough so that no data is
        hidden, then use the Sigmashet
        print to disc option to save the
        sheet."
1510 ENDPROC A

```


Fact-File

ABACUS Training	29 Okus Grove, Upper Stratton, Swindon, Wilts, SN2 6QA.
Academic Software	Sourby Old Farm, Timble, Otley, Yorks, LS21 2PW. (0943-88-628)
ACE Computing	27 Victoria Road, Cambridge, CB4 3BW. (0223-322559)
Blue Grey Software	28 Bobbitts Way, Wivenhoe, Essex, CO7 9NJ.
Brainsoft	22 Baker Street, London, W1M 1DF. (01-486-0321)
CCD Computer Services	71 Marlborough Park Avenue, Sidcup, Kent, DA15 9DL. (01-302-5427)
CJE Micros	78 Brighton Road, Worthing, W Sussex, BN11 2EN. (0903-213361)
Clares Micro Supplies	98 Middlewich Road, Rudheath, Northwich, Cheshire, CW9 7DA. (0606-48511)
Colton Software	149-151 St Neots Road, Hardwick, Cambridge, CB3 7QJ. (0954-211472)
Computer Concepts	Gaddesden Place, Hemel Hempstead, Herts, HP2 6EX. (0442-63933)
Computer Spares Ltd	61A Braemar Drive, Highcliffe, Dorset.
Contex Computing	15 Woodlands Close, Cople, Bedford, MK44 3UE. (02303-347)
Dudley Micro Services	30 Hadley Close, Netherton, Dudley, DY2 9JX. (0384-633142)
EMR Ltd	14 Mount Close, Wickford, Essex, SS11 8HG. (0702-335747)
Fairhurst Instruments	Dean Court, Woodford Road, Wilmslow, SK9 2LT. (0625-525-694)
GEM Electronics	17 Tandragee Road, Portadown, Craigavon, BT62 3BQ.
HopeSoft	Hope Cottage, Winterbourne, Newbury, Berks, RG16 8BB. (0635-248472)
Intelligent Interfaces Ltd	43b Wood Street, Stratford-upon-Avon, CV37 6JQ. (0789-415875)
Jansons	93 North Street, Stilton, Peterborough, PE7 3RR.
Julian Rockey	Brecklands, Broad Oak, Shrewsbury, SY4 3AH
Lingenuity	P.O.Box 10, Halesworth, Suffolk, IP19 0DX. (0986-85-476)
Minerva Systems	69 Sidwell Street, Exeter, EX4 6PH. (0392-37756)
Musbury Consultants	8 Fairhill, Helmsshore, Rossendale, Lancs, BB4 4JX. (0706-216701)
Pineapple Software	39 Brownlea Gardens, Seven Kings, Ilford, Essex, IG3 9NL. (01-599-1476)
SGB Computer Services	140 Disraeli Road, London, SW15 2DX. (01-874-5675)
Silicon Vision Ltd	Signal House, Lyon Road, Harrow, Middlesex, HA1 2AG. (01-422-2274)
Texellence	2 Greenhill Road, Coleraine, N Ireland, BT51 3JE.

Norwich Computer Services 18 Mile End Road, Norwich, NR4 7QY. (0603-507057)

Archive

Subscription Magazine and Support Group for *Archimedes* users

Archive Magazine contains:

- News
- Reviews
- Hints and Tips – a major feature
- Articles for Beginners
- The Latest Technical Information
- Program Listings
- Free Small Ad's Section
- HELP – Requested and Offered
- Contact Box – to help you form common interest groups

Eureka! – Bulletin Board

0603-250689 on 1200/75 or 300/300

- Very Latest News
- Down-load Software
- Mailbox Facilities
- Chat line

Technical Help Service (£8 / year)

A telephone hot-line service for immediate help with your technical problems. Any member can send written enquiries, but for a fast response use the THS!

Members' Discount: 7.5% off software from Computer Concepts, Minerva Systems and Clares Micros Supplies purchased through Norwich Computer Services.

Subscription: 12 issues £12.50 (UK)
Europe £18, Middle East £22,
America / Africa £25, Elsewhere £27.
Technical Help Service £8

N.B. All earlier issues have now been re-printed – you may back-date your subscription as far as issue 1 (October 1987) – to take advantage of this huge bank of information.

Archimedes is a trademark of Acorn Computers Ltd.

* Please send copies of *Archive* magazine for one year starting from
Volume 1 Issue _____

* Please enrol me on the Technical Help Service for one year. (£8)

I enclose a cheque for £ _____ payable to "Norwich Computer Services".

Name: _____

Address: _____

Postcode: _____

Norwich Computer Services, 18 Mile End Road, Norwich, NR4 7QY